AD-A066 344 EMMANUEL COLL BOSTON MASS F/G 9/2

INVESTIGATION OF MICROPULSATION ACTIVITY.(U)

NOV 78 W F BELLEW, M P HAGAN, R L VESPRINI F19628-76-C-0013

UNCLASSIFIED AFGL-TR-78-0312 NL

1 OF 2

AD
A066 344

AFGL-TR-78-0312

INVESTIGATIONS OF MICROPULSATION ACTIVITY

WILLIAM F. BELLEW
M. PATRICIA HAGAN
ROBERT L. VESPRINI

The Trustees of Emmanuel College
400 The Fenway
Boston, Massachusetts 02115

November 1978

Final Report: Period 15OCT75 - 30SEP78

Approved for public release; distribution unlimited.

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

MIL-STD-847A
31 January 1973

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (18) AFGL TR-78-0312 ✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A035941 - A056716 (6) INVESTIGATION OF MICROPULSATION ACTIVITY.		5. TYPE OF REPORT & PERIOD COVERED FINAL 15OCT75 — 30SEP78
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) WILLIAM F./BELLEW, M. PATRICIA/HAGAN, ROBERT L./VESPRINI		8. CONTRACT OR GRANT NUMBER(s) (15) F19628-76-C-0013 ✓
9. PERFORMING ORGANIZATION NAME AND ADDRESS Emmanuel College 400 The Fenway Boston MA 02115		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (16) 62101F 7601/08 01 (17) 08
11. CONTROLLING OFFICE NAME AND ADDRESS AIR FORCE GEOPHYSICS LABORATORY HANSCOM AFB MA 01731 CONTRACT MONITOR: ROBERT O. HUTCHINSON/PHG		12. REPORT DATE (11) NOV[redacted]R 1978
		13. NUMBER OF PAGES 125
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) (9) Final rept. 15 Oct 75-30 Sep 78		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

A - Approved for public release; distribution unlimited.

(12) 128p.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Archiving	Magnetic Field	CRT Display
Multiple Correlation	Spectral Analysis	Magnetograms
Magnetic Activity	Filtering	Data Storage
Micropulsations	Magnetometer Network	Statos Plot

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Programs, functions, and subroutines were designed to collect, archive, and provide user access to data received from the MAGAF data collection network. Subroutines were written to unpack a data frame from received data order to instrument order, and also to repack data into received → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

79 03 22 038

128950 LB

19. KEY WORDS (cont.)

Tektronix   Permutations   Data Retrieval

20. ABSTRACT (cont.)

order. A series of plotting routines were added to the system to enable plot files in Varian Dataplot format and to produce magnetograms from the magnetometer network. Mathematical analyses have been performed on the data, including analysis of magnetic pulsations and of micropulsation events.

CONTENTS

INTRODUCTION

The final report (AFCRL-TR-75-0588) prepared under a previous contract (F19628-73-C-0081) with Emmanuel College detailed the specifications of the mini-computer located at AFGL and installed for data reception in the MAGAF network.

The MAGAF Data Collection system software consists of various programs, functions, and subroutines designed to collect, archive, and provide user access to data received from the MAGAF data collection network. These routines operate under the VORTEX real-time, multitasked, disk operating system on a VARIAN V-72 minicomputer. (Multi-tasking is a scheme whereby many more or less independent programs compete for system resources under some kind of priority setup.) A brief description of MAGAF system flow follows.

Upon system initialization, a MAGAF initialization program is read in from the Foreground Library and executed. Since this program is on disk, it can be changed as required. Currently, this program checks the presence of the 5 kHz and 10 kHz signals to V-72 counter and interrupt clock and syncs the computer and its counter to the 10 second pulse from the master clock. At this point, control is returned to the VORTEX operating system with the core resident digital data unit, DDU, monitor active

The core resident monitor initiates a read into the first of two buffer areas and releases control to the VORTEX system. Upon receipt of a frame of data from the DDU, VORTEX reactivates the monitor which updates the pointer to the data for user programs and initiates a read into the second buffer area. This procedure continues using alternate buffer areas. Subroutines are available on the system library to access the data, convert the time code, and unpack the data into instrument order.

The data access monitor subroutine, when called by a user program, checks for a change in the data pointer provided by the DDU monitor. If there has been no change in this pointer, the program is deactivated for 50 milli-seconds and the pointer is checked again. If the pointer has changed, the frame is moved to the data area defined by the user program and the subroutine returns to the user program. It should be noted at this point that a change in this procedure is contemplated.

The interim archiving procedure takes the frame of data, compresses it when possible, stores it on disk in a circular buffer to maintain a short term history, and schedules the tape writing program when a buffer segment has been completed. Both programs, compression and tape writing are on disk and can therefore be modified.

I. MAGAF System

1. Library Functions and Archiving Procudure

(a) The library functions listed in Appendix A have been added to the system.

Function Name	Type	Calling Sequence	Result
IAND	Integer	I=IAND(J,K)	Returns logical product of J and K.
IOR	"	I=IOR(J,K)	Returns logical sum of J and K.
IXOR	"	I=IXOR(J,K)	Returns exclusive of J and K.
NOT	"	I=NOT(J)	Returns logical complement of J.
IBIT	"	I=IBIT(J,N)	$I=\begin{Bmatrix}0\\1\end{Bmatrix}$ if Nth bit (mod. 16) of J is $\begin{Bmatrix}0\\1\end{Bmatrix}$.
MASK	"	I=MASK(N)	Returns $\left\{\begin{matrix}\text{N leading ones}\\0\\\lvert N\rvert\text{ leading zeros}\end{matrix}\right\}$ if N $\left\{\begin{matrix}>0\\=0\\<0\end{matrix}\right\}$
ISHIFT	"	I=ISHIFT (J,N,ITP)	Returns J shifted N places according to ITP. ITP = 0 - Arithmetic left shift 1 - Logical rotation left 2 - Aritmetic right shift 3 - Logical right shift (0 fill)
ISET	"	I=ISET(N)	Returns a mask with bit N set to 1 (others 0). N out of range 0-15 undefined.
IRSET	"	I=IRSET(N)	Returns a mask with bit N set to 0 (others 1). N out of range 0-15 is undefined except, N = 17 returns left byte mask N = 18 " right byte mask

(b) MAGAF data collection interim archiving procedure:

The MAGAF interim archiving procedure consists of two programs which prepare the archive tape and a library subroutine and program for using previously prepared tapes. A description of the programs follows. Program listings are contained in Appendix B. The tape format is given in Appendix C.

I. Program ARCHIV monitors the incoming data, compresses it if possible, and buffers it out to disk. When the disk buffer is full, ARCHIV schedules program ARCTAP, which transfers the data to tape and changes tape units when the current unit is full.

II. Function I$TPY returns tape status and frame data to user programs, scheduling program DCMPRS when necessary. Program DCMPRS reads in one physical record from tape, decompresses the data, writes it out to disk, and returns to I$TPY tape status and the number of logical records contained to I$TPY.

The calling sequence for I$TPY is:

IST = I$TPY(IUNIT, IA)

where IST is tape status, IUNIT is the logical unit number of the tape, and IA is an array with dimension 245 which contains frame and status information.

	1	- good frame
	0	- device busy
IST =	-1	- end of tape
	-2	- end of file
	-3	- error

The format of IA is as described in AFCRL-TR-75-0588, with the following exceptions:

A. Data words and buffer contents are in the following form:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s	s	s	s	s	d	d	d	d	d	d	d	d	d	d	e

,

where s = sign bit (replicated)
d = data bit
e = error bit

B. IA(242) and IA(243) now contain frame time in seconds and milliseconds respectively, frame time being reset to zero every time real time reaches zero modulo ten seconds.

Appendix A

```
                                   VORTEX  DASMR                0014 HOURS
                      1          NAME     IAND
000000 000000 A       2 IAND     ENTR
                      3 *                          LOGICAL AND
                      4 *                          CALLING SEQUENCE: I=IAND(J,K)
000000                5 RETU     BES      0
                      6          EXT      $SE
000001 002000 A       7          CALL     $SE
000002 000000 E
000002                8 CALL     BES      0
000003 000002 A       9          DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A      10          LDAE*    CALL+2
000007 100004 R
000010 006157 A      11          ANAE*    CALL+3
000011 100005 R
000012 001000 A      12          RETU*    RETU
000013 100000 R
                     13          END
ENTRY NAMES
000000 R IAND
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IAND    000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                   VORTEX  DASMR                0014 HOURS

                      1          NAME     IOR
000000 000000 A       2 IOR      ENTR
                      3 *                          LOGICAL OR
                      4 *                          CALLING SEQUENCE:  I=IOR(J,K)
000000                5 RETU     BES      0
                      6          EXT      $SE
000001 002000 A       7          CALL     $SE
000002 000000 E
000002                8 CALL     BES      0
000003 000002 A       9          DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A      10          LDAE*    CALL+2
000007 100004 R
000010 006117 A      11          ORAE*    CALL+3
000011 100005 R
000012 001000 A      12          RETU*    RETU
000013 100000 R
                     13          END
ENTRY NAMES
000000 R IOR
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IOR     000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                    VORTEX  DASMR              0015 HOURS

                     1         NAME     IXOR
000000 000000 A      2 IXOR    ENTR
                     3 *                              EXCLUSIVE OR
                     4 *                            CALLING SEQUENCE:  I=IXOR(J,K)
000000               5 RETU    BSS      0
                     6         EXT      $SE
000001 002000 A      7         CALL     $SE
000002 000000 E
000002               8 CALL    BSS      0
000002 000002 A      9         DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A     10         LDAE*    CALL+2
000007 100004 R
000010 006137 A     11         ERAE*    CALL+3
000011 100005 R
000012 001000 A     12         RETU*    RETU
000013 100000 R
                    13         END
ENTRY NAMES
000000 R IXOR
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IXOR    000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                    VORTEX  DASMR              0015 HOURS

                     1         NAME     NOT
000000 000000 A      2 NOT     ENTR
                     3 *                              LOGICAL COMPLEMENT
                     4 *                               I=NOT(J)
000000               5 RETU    BSS      0
                     6         EXT      $SE
000001 002000 A      7         CALL     $SE
000002 000000 E
000002               8 CALL    BSS      0
000003 000001 A      9         DATA     1,0
000004 000000 A
000005 006017 A     10         LDAE*    CALL+2
000006 100004 R
000007 005211 A     11         CPA
000010 001000 A     12         RETU*    RETU
000011 100000 R
                    13         END
ENTRY NAMES
000000 R NOT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R NOT     000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                              VORTEX  DASMR                    0015 HOURS

                      1           NAME     IBIT
000000 000000 A       2 IBIT      ENTR
                      3 *                              J=IBIT(I,N)
                      4 *                              PETURNS IN J THE VALUE(0,1) OF T-E
                      5 *                              OF I;  N IS TAKEN MODULO 16.
000000                6 PETU      BES      0
                      7           EXT      $SE
000001 002000 A       8           CALL     $SE
000002 000000 E
000002                9 CALL      BES      0
000003 000002 A      10           DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A      11           LDAE*    CALL+3
000007 100005 R
000010 074011 A      12           STX      STX+1
000011 150472 A      13           ANA      0472                MASK 017
000012 005014 A      14           TAX
000013 015421 A      15           LDA      0421,1
000014 006157 A      16           ANAE*    CALL+2
000015 100004 R
000016 001010 A      17           JAZ      *+3
000017 000021 R
000020 005101 A      18           INCR     01
000021 006030 A      19 STX       LDXI     *
000022 000021 R
000023 001000 A      20           RETU*    PETU
000024 100000 R
                     21           END
ENTRY NAMES
000000 R IBIT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IBIT    000000 R RETU
000021 R STX
  0 ERFORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR                  0015 HOURS
                          1
                          2
                          3
                          4             NAME     MASK
000000 000000 A           5 MASK        ENTR
                          6 *                                       I=MASK(N) GIVES A MASK OF N L
                          7 *                                       IF N IS POSITIVE, N 0'S FOLL
                          8 *                                       IF N IS NEGATIVE
000000                    9 RETU        BES      0
                         10             EXT      $SE
000001 002000 A          11             CALL     $SE
000002 000000 E
000002                   12 CALL        BES      0
000003 000001 A          13             DATA     1.0
000004 000000 A
000005 006017 A          14             LDAE*    CALL+2
000006 100004 R
000007 001010 A          15             JAZ*     RETU
000010 100000 R
000011 074017 A          16             STX      STX+1
000012 005004 A          17             TIX
000013 001002 A          18             JAP      LEAD
000014 000021 R
000015 005211 A          19             CPA
000016 005111 A          20             IAR
000017 006030 A          21             LDXI     041                  MASK DISPLACEMENT
000020 000041 A
000021 005311 A          22 LEAD        DAR
000022 150472 A          23             ANA      0472                MASK 017
000023 114010 A          24             ORA      INSTR
000024 054001 A          25             STA      *+2
000025 010440 A          26             LDA      0440                  SIGN BIT
000026 004300 A          27             ASRA     0
000027 135420 A          28             ERA      0420,1                  EITHER 0 OR 177777
000030 006030 A          29 STX         LDXI     *
000031 000030 R
000032 001000 A          30             RETU*    RETU
000033 100000 R
000034 004300 A          31 INSTR       ASRA     0
                         32             END
ENTRY NAMES
000000 R MASK
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000034 R INSTR   000021 R LEAD
000000 R MASK    000000 R RETU    000030 R STX
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR              0015 HOURS

                              1
                              2          NAME     ISHIFT
000020 000000 A               3 ISHIFT ENTR
                              4 *                            I=ISHIFT(IWHAT,N,ITYPE)
                              5 *                               SHIFTS IWHAT N PLACES; TYPE OF SHI
                              6 *                            ITYPE  0=ARITHMETIC SHIFT LEFT
                              7 *                                   1=LOGICAL ROTATION LEFT(SIGN
                              8 *                                   2=ARITMETIC SHIFT RIGHT
                              9 *                                  3=LOGICAL SHIFT RIGHT(SHIFTED
000000                       10 RETU    BES       0
                             11         EXT       $SE
000001 002000 A              12         CALL      $SE
000002 000000 E
000002                       13 CALL    BES       0
000003 000003 A              14         DATA      3,0,0,0
000004 000000 A
000005 000000 A
000006 000000 A
000007 064016 A              15         STB       STB+1
000010 006017 A              16         LDAE*     CALL+4
000011 100006 R
000012 006027 A              17         LDBE*     CALL+3
000013 100005 R
000014 004053 A              18         LRLB      11
000015 004542 A              19         LLSR      2
000016 006010 A              20         LDAI      17
000017 000021 A
000020 004447 A              21         LLRL      7
000021 054002 A              22         STA       *+3
000022 006017 A              23         LDAE*     CALL+2
000023 100004 R
000024 005000 A              24         NOP
000025 006020 A              25 STB     LDBI      *
000026 000025 R
000027 001000 A              26         RETU*     RETU
000030 100000 R
                             27         END
ENTRY NAMES
000000 R ISHIFT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE      000002 R CALL     000000 R ISHIFT 000000 R RETU
000025 R STB
  0 ERRORS ASSEMBLY COMPLETE
```

```
                              VORTEX  DASMP              0015 HOURS
                    1
                    2
                    3
                    4              NAME     ISET
000000 000000 A     5 ISET         ENTR
                    6 *                                J=ISET(N)
                    7 *                                  RETURNS A BIT MASK WITH BIT N SE
                    8 *                                  1 (OTHERS 0). N OUT OF RANGE 0-1
000000              9 RETU         BES      0
                   10              EXT      $SE
000001 002000 A    11              CALL     $SE
000002 000000 E
000002             12 CALL         BES      0
000003 000001 A    13              DATA     1,0
000004 000000 A
000005 074004 A    14              STX      STX+1
000006 006037 A    15              LDXE*    CALL+2
000007 100004 R
000010 015421 A    16              LDA      0421,1
000011 006030 A    17 STX          LDXI     *
000012 000011 R
000013 001000 A    18              RETU*    RETU
000014 100000 R
                   19              END
ENTRY NAMES
000000 R ISET
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R ISET    000000 R RETU
000011 R STX
  0 ERRORS ASSEMBLY COMPLETE
                              VORTEX  DASMR              0015 HOURS
                    1              NAME     IRSET
000000 000000 A     2 IRSET        ENTR
                    3 *                                J=IRSET(N)
                    4 *                                  RETURNS A BIT MASK WITH BIT N SE
                    5 *                                  0 (OTHERS 1). N OUT OF RANGE 0-1
                    6 *                                  EXCEPT 17 GIVES LEFT BYTE MASK. 1
000000              7 RETU         BES      0
                    8              EXT      $SE
000001 002000 A     9              CALL     $SE
000002 000000 E
000002             10 CALL         BES      0
000003 000001 A    11              DATA     1,0
000004 000000 A
000005 074004 A    12              STX      STX+1
000006 006037 A    13              LDXE*    CALL+2
000007 100004 R
000010 015441 A    14              LDA      0441,1
000011 006030 A    15 STX          LDXI     *
000012 000011 R
000013 001000 A    16              RETU*    RETU
000014 100000 R
                   17              END
ENTRY NAMES
000000 R IRSET
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IRSET   000000 R RETU
000011 R STX
  0 ERRORS ASSEMBLY COMPLETE
```

Appendix B

Program ARCHIV

```
        000422 A    1 TWO    EQU     0422
        000423 A    2 FOUR   EQU     0423
        000424 A    3 EIGHT  EQU     0424
        000466 A    4 SIX    EQU     0466
        000467 A    5 SEVEN  EQU     0467
        000443 A    6 MS     EQU     0443                MINUS FIVE
        000465 A    7 FIVE   EQU     0465
        000464 A    8 THREE  EQU     0464
        000472 A    9 FIFTH  EQU     0472
        000002 A   10 B      EQU     2
        000001 A   11 X      EQU     1
        000025 A   12 TUNIT  EQU     21
        000041 A   13 DUNIT  EQU     33
                   14 ARC    REW     FCB,DUNIT
000000 006505 A
000001 000000 E
000002 100000 A
000003 001441 A
000004 000175 R
000005 000000 A
000006 002000 A
000007 006010 A    15        LDAI    TUNIT
000010 000025 A
000011 050005 A    16        STA     5
000012 006010 A    17        LDAI    DUNIT
000013 000041 A
000014 050006 A    18        STA     6
000015 006010 A    19        LDAI    FCB
000016 000175 P
000017 050004 A    20        STA     4
000020 002000 A    21 ARCH1  CALL    ARCHIV
000021 000211 R
000022 124147 A    22        ADD     NWD1
000023 005012 A    23        TAB
000024 144146 A    24        SUB     DCB
000025 001004 A    25        JAN     ADD
000026 000067 P
000027 005301 A    26        DECP    01
000030 002000 A    27        CALL    OUT
000031 000075 P
000032 014145 A    28        LDA     FCB+3
000033 124154 A    29        ADD     TWOTWO
000034 144152 A    30        SUB     NSECT
000035 054143 A    31        STA     FCB+3
000036 005001 A    32        TZA
000037 054147 A    33        STA     NSECT
000040 014131 A    34        LDA     NWD1
                   35        SCHED   31,1,106,'F','AR','CT','AP'
000041 006505 A
000042 000000 E
000043 010137 A
000044 140152 A
000045 140722 A
```

```
                                             VORTEX  DASMR                0000 HOURS

000046 141724 A
000047 140720 A
000050 024524 A     36            LDB      NWDS
000051 014130 A     37            LDA      FCB+3
000052 121135 A     38            ADD      TWOTWO
000053 144135 A     39            SUB      FCB+4
000054 001004 A     40            JAN      ADD
000055 000067 R
000056 001010 A     41            JAZ      ADD
000057 000067 R
                    42            REW      FCB,DUNIT
000060 006505 A
000061 000001 E
000062 100000 A
000063 001441 A
000064 000175 R
000065 000000 A
000066 000000 A
000067 064102 A     43 ADD        STB      NWD1
000070 014504 A     44            LDA      NWDS
000071 002000 A     45            CALL     OUT
000072 000075 R
000073 001000 A     46            JMP      ARCH1
000074 000020 R
                    47 *
                    48 *
                    49 *
000075 000000 A     50 OUT        ENTR
000076 054464 A     51            STA      TMP
000077 006057 A     52            STAE*    ABUF
000100 100171 R
000101 001004 A     53            JAN      WRITE
000102 000115 R
000103 014453 A     54            LDA      HDAD
000104 054454 A     55            STA      STRT1
000105 014061 A     56 LFT1       LDA      LEFT
000106 005014 A     57            TAX
000107 144453 A     58            SUB      TMP
000110 005311 A     59            DAR
000111 001002 A     60            JAP      FILL
000112 000144 R
000113 006500 A     61            JSR      FILLIN,B
000114 000156 R
                    62 WRITE      WRITE    FCB,DUNIT
000115 006505 A
000116 000061 E
000117 100000 A
000120 000441 A
000121 000175 R
000122 000000 A
000123 000000 A
000124 044062 A     63            INR      NSECT
000125 014453 A     64            LDA      STRT1
```

```
000126 1.4040 A    69           ADD    LEFT
000127 054411 A    60           STA    STRT1
000130 014432 A    62           LDA    TMP
000131 1440.. A    68           SUB    LEFT
000132 024041 A    69           LDB    DCB+1
000133 064035 A    70           STB    ABUF
000134 024033 A    71           LDB    D120
000135 064031 A    72           STB    LEFT
000136 054424 A    73           STA    TMP
000137 005311 A    74           DAP
000140 001002 A    75           JAP    LFT1
000141 000105 R
000142 001000 A    76           RETU*  OUT
000143 100075 R
000144 005111 A    77 FILL      IAR
000145 054021 A    78           STA    LEFT
000146 034414 A    79           LDX    TMP
000147 006506 A    80           JSP    FILLIN,B
000150 000156 R
000151 014411 A    81           LDA    TMP
000152 124016 A    82           ADD    ABUF
000153 054015 A    83           STA    ABUF
000154 001000 A    84           RETU*  OUT
000155 100075 R
                   85 *
                   86 *
                   87 *
000156 005344 A    88 FILLIN    DXR
000157 006215 A    89           LDAE*  STRT1,X,0200
000160 100561 R
000161 006255 A    90           STAE*  ABUF,X,0200
000162 100171 R
000163 001046 A    91           JXNZ   FILLIN
000164 000156 R
000165 006706 A    92           IJMP   0,B
000166 000000 A
                   93 *
                   94 *
                   95 *
000167 000170 A    96 LEFT      DATA   120
000170 000170 A    97 D120      DATA   120
000171 001253 R    98 ABUF      DATA   BUFR
000172 000000 A    99 HWD1      DATA   0
                  100 DCB       DCB    2560,BUFR
000173 005000 A
000174 001253 R
000175 000000 A
000175            101 FCB       ORG    *-1
                  102           FCB    120,BUFR,3
000175 000170 A
000176 001253 R
000177 001400 A
000200 000000 A
```

```
000201 000000 A
000202 000000 A
000203 000000 A
000204 000000 A
000205 000000 A
000206 000000 A
000207 000000 A   103 NSECT  DATA    0
000210 000026 A   104 TWOTWO DATA    22
                  105 *
                  106 *
                  107 *
       000211 R   108 ARCHIV EQU     *
000211 000212 R   109        DATA    *+1
                  110 DELAY  DELAY   1
000212 006505 A
000213 000042 E
000214 001100 A
000215 000001 A
000216 000000 A
000217 010002 A   111 DELY1  LDA     2
000220 144351 A   112        SUB     CUR
000221 001010 A   113        JAZ     DELAY
000222 000212 R
000223 030002 A   114        LDX     2
000224 074345 A   115        STX     CUR
000225 015364 A   116        LDA     244,X
000226 054346 A   117        STA     NWDS
000227 015360 A   118        LDA     240,X
000230 054345 A   119        STA     STAT
000231 015363 A   120        LDA     243,X
000232 054356 A   121        STA     BUFF
000233 020003 A   122        LDB     3              TIME
000234 005001 A   123        TZA
000235 004441 A   124        LLRL    1
000236 004141 A   125        LSRB    1
000237 006170 A   126        DIVI    5000
000240 011610 A
000241 064345 A   127        STB     SEC            SECONDS
000242 120464 A   128        ADD     THREE          ROUND
000243 004560 A   129        LLSR    16
000244 170465 A   130        DIV     FIVE
000245 064342 A   131        STB     MILLI          MILLISECONDS
                  132        EXT     #UNPK
000246 000000 A   133        CALL    #UNPK,(CUR)*,SCY
000247 000000 E
000250 100572 P
000251 000012 R
000252 034324 A   134        LDX     NWDS
000253 005144 A   135        IXR
000254 005344 A   136 DXR1   DXR
000255 006015 A   137        LDAE    BUFF,X
000256 000011 R
000257 004601 A   138        LASR    1
```

```
000260 004304 A  139           ASRA   4
000261 004401 A  140           LASL   1
000262 006055 A  141           STAE   BUFF,X
000263 000611 R
000264 001046 A  142           JXNZ   DXR1
000265 000254 R
000266 014207 A  143           LDA    STAT
000267 001002 A  144           JAP    PACK
000270 000307 R
000271 006010 A  145           LDAI   SEC
000272 000607 P
000273 054265 A  146           STA    STRT1
000274 006010 A  147           LDAI   IPACK
000275 000577 R
000276 054261 A  148           STA    START
000277 010464 A  149           LDA    THREE
000300 124274 A  150           ADD    NWDS
000301 054003 A  151           STA    *+3
000302 006505 A  152           JSR    MOV,X
000303 000540 P
000304 005000 A  153           NOP
000305 001000 A  154           JMP    RET1
000306 000353 P
       000307 P  155 PACK      EQU    *
                 156           EXT    SNAP
000307 006010 A  157           LDAI   SCK
000310 000612 P
000311 054240 A  158           STA    START
000312 054246 A  159           STA    STRT1
000313 006010 A  160           LDAI   IPK1
000314 000577 R
000315 054244 A  161           STA    IPKWRT
000316 006010 A  162           LDAI   50
000317 000062 A
000320 002000 A  163           CALL   CMPRS
000321 000360 R
000322 002000 A  164           CALL   CMPRS
000323 000360 P
000324 002000 A  165           CALL   CMPRS
000325 000360 P
000326 006010 A  166           LDAI   10
000327 000012 A
000330 002000 A  167           CALL   CMPRS
000331 000360 P
000332 002000 A  168           CALL   CMPRS
000333 000360 P
000334 002000 A  169           CALL   CMPRS
000335 000360 P
000336 006505 A  170           JSR    MOV,X
000337 000540 P
000340 000011 A  171           DATA   9
000341 006010 A  172           LDAI   23
000342 000027 A
```

```
000343 002000 A   173            CALL     CMPRS
000344 000360 R
000345 010472 A   174            LDA      FIFTN
000346 002000 A   175            CALL     CMPRS
000347 000360 R
000350 006505 A   176            JSR      MOV,X
000351 000540 R
000352 000015 A   177            DATA     13
000353 014204 A   178 RET1       LDA      START
000354 144202 A   179            SUB      HDAD
000355 054217 A   180            STA      NWDS
000356 001000 A   181            RETU*    ARCHIV
000357 100211 R
                  182 *
                  183 *
                  184 *
000360 000000 A   185 CMPRS      ENTR
000361 005311 A   186            DAR
000362 054144 A   187            STA      N
000363 006027 A   188            LDBE*    STRT1
000364 100561 R
000365 002000 A   189            CALL     INSRT
000366 000532 R
000367 005311 A   190            DAR
000370 005014 A   191            TAX
000371 124167 A   192            ADD      STRT1
000372 005012 A   193            TAB
000373 005001 A   194            TZA
000374 054156 A   195            STA      TMP
000375 016001 A   196 LDA        LDA      1,B
000376 146000 A   197            SUB      0,B
000377 004301 A   198            ASFA     1
000400 006055 A   199            STAE     TEMP,X
000401 001172 R
000402 001002 A   200            JAP      SUB
000403 000410 R
000404 005211 A   201            CPA
000405 001016 A   202            JANZ     *+3
000406 000410 R
000407 005111 A   203            IAR
000410 144152 A   204 SUB        SUB      TMP
000411 001004 A   205            JAN      *+4
000412 000415 R
000413 134147 A   206            ADD      TMP
000414 054140 A   207            STA      TMP
000415 005322 A   208            DBR
000416 005344 A   209            DXR
000417 001040 A   210            JXNZ     LDA
000420 000375 R
000421 010443 A   211            LDA      M5                        MINUS FIVE
000422 024140 A   212            LDB      TMP
000423 001020 A   213            JBZ      DONE
000424 000441 R
```

```
000425 005111 A   214 IAR    IAR
000426 001010 A   215        JAZ     DN1
000427 000435 R
000430 001020 A   216        JBZ     DONE
000431 000441 R
000432 004101 A   217        ASRB    1
000433 001000 A   218        JMP     IAR
000434 000425 R
000435 005111 A   219 DN1    IAR
000436 004102 A   220        ASRB    2
000437 001036 A   221        JBNZ    TWEL
000440 000521 R
000441 120465 A   222 DONE   ADD     FIVE
000442 006057 A   223        STAE*   IPKWRT
000443 100563 R
000444 054121 A   224        STA     NBITS
000445 001010 A   225        JAZ     ZBITS
000446 000510 R
000447 005001 A   226        TZA
000450 054114 A   227        STA     N1
000451 024116 A   228        LDB     TWELVE
000452 174113 A   229        DIV     NBITS
000453 064113 A   230        STB     NTOWD
000454 014114 A   231        LDA     LLSR
000455 124110 A   232        ADD     NBITS
000456 054007 A   233        STA     LLS
000457 006010 A   234        LDAI    TEMP
000460 001172 R
000461 054100 A   235        STA     TMP1
000462 034104 A   236        LDX     NTOWD
000463 006017 A   237 NXT1   LDAE*   TMP1
000464 100564 R
000465 044076 A   238        INR     TMP1
000466 004540 A   239 LLS    LLSR    0
000467 005344 A   240        DXR
000470 001046 A   241        JXNZ    NXT1
000471 000463 R
000472 004544 A   242        LLSR    4
000473 002000 A   243        CALL    INSRT
000474 [illegible] R
000475 014067 A   244        LDA     N1
000476 124070 A   245        ADD     NTOWD
000477 054065 A   246        STA     N1
000500 144066 A   247        SUB     N
000501 001004 A   248        JAN     NXT1-1
000502 000462 R
000503 034065 A   249        LDX     N
000504 [illegible] A   250        LDBE*   STRT1,X,0200
000505 100561 R
000506 002000 A   251        CALL    INSRT
000507 [illegible] R
000510 014060 A   252 ZBITS  LDA     STRT1
000511 [illegible] A   253        ADD     8
```

```
000512 005111 A   254            IAR
000513 054045 A   255            STA      STRT1
000514 014012 A   256 LODX       LDA      N
000515 005111 A   257            IAR
000516 044043 A   258            INR      IPKWRT
000517 001000 A   259            RETU*    CMPPS
000520 100360 R
000521 034046 A   260 TWEL       LDX      TWELVE
000522 006077 A   261            STXE*    IPKWRT
000523 100562 R
000524 044034 A   262            INR      STRT1
000525 006505 A   263            JSR      MOV,X
000526 000540 R
000527 000000 A   264 N          DATA     0
000530 001000 A   265            JMP      LODX
000531 000514 R
000532 000000 A   266 INSRT      ENTR
000533 006067 A   267            STBE*    START
000534 100560 R
000535 044022 A   268            INR      START
000536 001000 A   269            RETU*    INSRT
000537 100532 R
                  270 *
                  271 *
                  272 *
000540 025000 A   273 MOV        LDB      0,X
000541 005144 A   274            IXR
000542 074013 A   275            STX      JUMP+1
000543 034014 A   276            LDX      START
000544 005322 A   277 DBR1       DBR
000545 006017 A   278            LDAE*    STRT1
000546 100561 R
000547 044011 A   279            INR      STRT1
000550 055000 A   280            STA      0,X
000551 005144 A   281            IXR
000552 001000 A   282            JBNZ     DBR1
000553 000544 R
000554 074002 A   283            STX      START
000555 001000 A   284 JUMP       JMP      *
000556 000555 R
                  285 *
                  286 *
                  287 *
000557 000573 R   288 HDAD       DATA     HEAD
000560 000000 A   289 START      DATA     0
000561 000000 A   290 STRT1      DATA     0
000562 000000 A   291 IPKWRT     DATA     0
000563 000000 A   292 TMP        DATA     0
000564 000000 A   293 TMP1       DATA     0
000565 000000 A   294 QI         DATA     0
000566 000000 A   295 ABITS      DATA     0
000567 000000 A   296 NEWHD      DATA     0
000570 000014 A   297 TWELVE     DATA     12
```

```
                                    VORTEX  DASMR                0000 HOU

000571 004540 A  298 LLSR    LLSR     0
000572 000000 A  299 CUR     DATA     0
                 300 *
                 301 *
                 302 *
000573 000001 A  303 HEAD    DATA     1,0
000574 000000 A
000575 000000 A  304 NWDS    DATA     0
000576           305 STAT    BSS      1
000577 000000 A  306 IPACK   DATA     0
       000577 R  307 IPK1    EQU      IPACK
000600           308         BSS      7
000607           309 SEC     BSS      1
000610           310 MILLI   BSS      1
000611           311 BUFF    BSS      1
000612           312 SCX     BSS      240
001172           313 TEMP    BSS      49
001253           314 BUFR    BSS      120
       000000 R  315         END      ARC
ENTRY NAMES
EXTERNAL NAMES
000247 E $UNPK  000000 E SNAP   000213 E V$EXEC 000116 E V$IOC
SYMBOLS
000247 E $UNPK  000171 R ABUF   000067 R ADD    000000 R ARC
000020 R ARCH1  000211 R ARCHIV 000002 A B      000511 R BUFF
001253 R BUFR   000360 R CMPRS  000572 R CUR    000170 R D120
000544 R DBR1   000173 R DCB    000212 R DELAY  000217 R DELY1
000435 R DN1    000441 R DONE   000041 A DUNIT  000254 R DXR1
000424 A EIGHT  000175 R FCB    000472 A FIFTH  000144 R FILL
000156 R FILLIN 000465 A FIVE   000463 A FOUR   000557 R HBAD
000573 R HEAD   000425 R INB    000533 R INSRT  000577 R IPACK
000577 R IPK1   000562 R IPKWRT 000555 R JUMP   000375 R LDA
000167 R LEFT   000105 R LFT1   000400 R LLS    000571 R LLSR
000514 R LODX   000443 A M5     000610 R MILLI  000540 R MOV
000527 R N      000565 R N1     000566 R NBITS  000207 R NSECT
000567 R NTOWD  000172 R NWD1   000575 R NWDS   000463 R NXT1
000075 R OUT    000307 R PACK   000353 R RET1   000612 R SCX
000607 R SEC    000467 A SEVEN  000466 A SIX    000000 E SNAP
000560 R START  000576 R STAT   000561 R STRT1  000410 R SUB
001172 R TEMP   000464 A THREE  000563 R TMP    000564 R TMP1
000025 A TUNIT  000521 R TWEL   000570 R TWELVE 000420 A TWO
000210 R TWOTWO 000213 E V$EXEC 000116 E V$IOC  000115 R WRITE
000001 A X      000510 R ZBITS
  0 ERRORS ASSEMBLY COMPLETE
```

Program ARCTAP

```
                                   VORTEX  DASMR                  0000 HOURS

       000002 A    1 B       EQU     2
       000462 A    2 LHW     EQU     0462
       000005 A    3 TUNIT   EQU     5
       000006 A    4 DUNIT   EQU     6
       000004 A    5 AFCB    EQU     4
000000 005111 A    6 START   IAP
000001 054116 A    7         STA     FCB
000002 010006 A    8         LDA     DUNIT
000003 124007 A    9         ADD     REW+3
000004 054006 A   10         STA     REW+3
000005 010006 A   11         LDA     DUNIT
000006 124020 A   12         ADD     READ+3
000007 054017 A   13         STA     READ+3
                  14 REW     REW     FCB
000010 006505 A
000011 000000 E
000012 100000 A
000013 001400 A
000014 000120 R
000015 000000 A
000016 000000 A
000017 020004 A   15         LDB     AFCB
000020 016003 A   16         LDA     3,B
000021 006140 A   17         SUBI    22
000022 000026 A
000023 054077 A   18         STA     FCB+3
                  19 READ    READ    FCB,,,3
000024 006505 A
000025 000011 E
000026 100000 A
000027 030000 A
000030 000120 R
000031 000000 A
000032 000000 A
000033 002000 A   20         CALL    WRITRY
000034 000055 R
000035 001010 A   21         JAZ     EXIT
000036 000052 R
000037 006010 A   22 FOR3    LDAI    43
000040 000053 A
000041 140005 A   23         SUB     TUNIT
000042 050005 A   24         STA     TUNIT
000043 002000 A   25         CALL    WRITRY
000044 000055 R
000045 001010 A   26         JAZ     EXIT
000046 000052 R
000047 000000 A   27         HLT
000050 001000 A   28         JMP     FOR3
000051 000037 R
                  29 EXIT    EXIT
000052 006500 A
000053 000000 E
000054 000200 A
```

```
                                   VORTEX  DASMR                 0000 HOURS

000055 000000 A    30 WRITRY ENTR
000056 014003 A    31        LDA     WRITE+3
000057 150462 A    32        ANA     LAW
000060 130005 A    33        ADD     TUNIT
000061 054003 A    34        STA     WRITE+3
                   35 WRITE  WRITE   FCB,21,,4
000062 006505 A
000063 000055 E
000064 100000 A
000065 040425 A
000066 000120 R
000067 000000 A
000070 000000 A
                   36        STAT    WRITE,NEXT,NEXT,EOT,NEXT
000071 006505 A
000072 000000 E
000073 000062 R
000074 000100 R
000075 000100 R
000076 000103 R
000077 000100 R
000100 005001 A    37 NEXT   TZA
000101 001000 A    38        JMP*    WRITRY
000102 100055 R
000103 014006 A    39 EOT    LDA     SPEC+3
000104 150462 A    40        ANA     LAW
000105 130005 A    41        ADD     TUNIT
000106 054003 A    42        STA     SPEC+3
                   43 SPEC   SREC    FCB,21,,1
000107 006505 A
000110 000062 E
000111 100000 A
000112 012025 A
000113 000120 R
000114 000000 A
000115 000000 A
000116 001000 A    44        JMP*    WRITRY
000117 100055 R
                   45 FCB    FCB     2560,BUFR,3
000120 005000 A
000121 000132 R
000122 001400 A
000123 000000 A
000124 000000 A
000125 000000 A
000126 000000 A
000127 000000 A
000130 000000 A
000131 000000 A
000132             46 BUFR   BSS     2560
       000000 R    47        END     START
ENTRY NAMES
EXTERNAL NAMES
                                   VORTEX  DASMR                 0000 HOURS

000053 E V$EXEC 000110 E V$IOC  000072 E V$IOST
SYMBOLS
000004 A AFCB   000002 A B      000132 R BUFR   000006 A DUNIT
000103 R EOT    000052 R EXIT   000120 R FCB    000037 R FOP3
000462 A LAW    000100 R NEXT   000024 R READ   000010 R REW
000107 R SPEC   000000 R START  000005 A TUNIT  000053 E V$EXEC
000110 E V$IOC  000072 E V$IOST 000062 R WRITE  000055 R WRITRY
  0 ERRORS ASSEMBLY COMPLETE
```

Function I$TPY

```
                                   VORTEX  DASMR                    0001 HOURS
                         1            EXT      $SE,$BUFF
                         2            NAME     I$TPY
       000001 A          3 X          EQU      1
       000002 A          4 B          EQU      2
000000 014106 A          5 START      LDA      INUM
000001 024076 A          6            LDB      IA
000002 054110 A          7            STB      FCB+1
000003 001016 A          8            JANZ     NOSCED
000004 000030 R
000005 006020 A          9            LDBI     CALSEQ
000006 000103 R
                        10            SCHED    5,1,106,'F','DC','MP','RS'
000007 006505 A
000010 000000 E
000011 010105 A
000012 143152 A
000013 142303 A
000014 146720 A
000015 151323 A
000016 014071 A         11            LDA      IST
000017 001004 A         12            JAN      DECRM
000020 000066 R
                        13            REW      FCB,10
000021 006505 A
000022 000000 E
000023 100000 A
000024 001412 A
000025 000112 R
000026 000000 A
000027 000000 A
                        14 NOSCED     READ     FCB,10,,3
000030 006505 A
000031 000022 E
000032 100000 A
000033 030012 A
000034 000112 R
000035 000000 A
000036 000000 A
000037 024040 A         15            LDB      IA
000040 016364 A         16            LDA      244,B
000041 001010 A         17            JAZ      WIP
000042 000050 R
000043 001002 A         18            JAP      DECRM-1
000044 000065 R
000045 044047 A         19            INR      FCB+3
000046 001000 A         20            JMP      NOSCED
000047 000030 R
000050 005021 A         21 WIP        TBA
000051 006140 A         22            SUBI     5
000052 000005 A
000053 054041 A         23            STA      FCB+3
                        24            WRITE    FCB,10,,3
000054 006505 A
```

```
000055 000031 E
000056 100000 A
000057 030412 A
000060 000113 R
000061 000000 A
000062 000000 A
000063 001000 A    25          JMP     DECRM
000064 000066 R
000065 044027 A    26          INR     FCB+3
000066 014020 A    27 DECRM    LDA     INUM
000067 005311 A    28          DAR
000070 054016 A    29          STA     INUM
000071 014016 A    30          LDA     IST
000072 001000 A    31          JMP
000073 000000 A
000073             32 I$TRY    BES     0
000074 002000 A    33          CALL    $SE
000075 000000 E
000076 000002 A    34          DATA    2
000077 000000 A    35 IUNIN    DATA    0
000100 000000 A    36 IA       DATA    0
000101 001000 A    37          JMP     START
000102 000000 R
000103 100077 R    38 CALSEQ   DATA    (IUNIN)*,IST,IUNOT,INUM
000104 000110 R
000105 000111 R
000106 000107 R
000107 000000 A    39 INUM     DATA    0
000110 000000 A    40 IST      DATA    0
000111 000012 A    41 IUNOT    DATA    10
                   42 FCB      FCB     245,0,0
000112 000365 A
000113 000000 A
000114 000000 A
000115 000000 A
000116 000000 A
000117 000000 A
000120 000000 A
000121 000000 A
000122 000000 A
000123 000000 A
                   43          END
ENTRY NAMES
000073 R I$TRY
EXTERNAL NAMES
000000 E $BUFF   000075 E $SE     000010 E V$EXEC 000055 E V$IOC
SYMBOLS
000000 E $BUFF   000075 E $SE     000002 A B      000103 R CALSEQ
000066 R DECRM   000112 R FCB     000073 R I$TRY  000100 R IA
000107 R INUM    000110 R IST     000077 R IUNIN  000111 R IUNOT
000030 R NOSCED  000000 S START   000010 E V$EXEC 000055 E V$IOC
000050 R WIP     000001 A I
   0 ERRORS ASSEMBLY COMPLETE
```

Program DCMPRS

```
                      1          EXT     $SE
       000001 A       2 X        EQU     1
       000420 A       3 ZERO     EQU     0420
       000423 A       4 FOUR     EQU     0423
       000002 A       5 B        EQU     2
                      6          EXT     $$BUFF
000000 002000 A       7 START    CALL    $$BUFF,(CALSEQ)*,BUFR,D2560,ZERO,FOUR
000001 000000 E
000002 100037 R
000003 000056 R
000004 000055 R
000005 000420 A
000006 000423 A
000007 034030 A       8          LDX     CALSEQ+1
000010 055000 A       9          STA     0,X
000011 001004 A      10          JAN     ERROR
000012 000025 R
000013 014042 A      11          LDA     BUFR
000014 001010 A      12          JAZ     START
000015 000000 R
000016 064036 A      13          STB     NWDS
                     14          EXT     DCMP
000017 002000 A      15          CALL    DCMP,(CALSEQ+2)*,(CALSEQ+3)*,BUFR,NWDS
000020 000000 E
000021 100041 R
000022 100042 R
000023 000056 R
000024 000055 R
                     16 ERROR    EXIT
000025 006505 A
000026 000000 E
000027 000200 A
000030 064003 A      17 ENTRY    STB     *+3
000031 001000 A      18          JMP     *+3
000032 000034 R
000033 000045 R      19          DATA    TST
000034 002000 A      20          CALL    $SE
000035 000000 E
000036 000004 A      21          DATA    4
000037 000000 A      22 CALSEQ   DATA    0,0,0,0
000040 000000 A
000041 000000 A
000042 000000 A
000043 001000 A      23          JMP     START
000044 000000 R
       000045 R      24 TST      EQU     *
                     25          DUP     4
000045 000051 R      26          DATA    *+4
000046 000052 R      26          DATA    *+4
000047 000053 R      26          DATA    *+4
000050 000054 R      26          DATA    *+4
000051 0000[illegible] A      27 TST1     DATA    [illegible],0,[illegible],0
000052 000000 A
```

```
                                VORTEX  DASMP          0000 HOURS
000053 000041 A
000054 000000 A
000055 005000 A     28 D2560   DATA      2560
       000055 R     29 NWDS    EQU       D2560
000056              30 BUFR    BSS       2560
       000030 R     31         END       ENTRY
ENTRY NAMES
EXTERNAL NAMES
000001 E IIBUFF 000035 E $SE     000020 E DCMP    000026 E V$EXEC
SYMBOLS
000001 E IIBUFF 000035 E $SE     000002 A B       000056 R BUFR
000037 R CALSEO 000055 R D2560   000020 E DCMP    000030 R ENTRY
000025 R ERROR  000423 A FOUR    000055 R NWDS    000000 R START
000045 R TST    000051 R TST1    000026 E V$EXEC  000001 A X
000420 A ZERO
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR    $BUFF      0000 HOURS

                          1            TITLE    $BUFF
                          2            NAME     $$BUFF
                          3            NAME     LRECL
                          4            EXT      $SE
       000421 A           5 ONE        EQU      0421
000000 000000 A           6 $$BUFF     ENTR
000001 002000 A           7            CALL     $SE
000002 000000 E
000003 000005 A           8            DATA     5
000004                    9 UNIT       BSS      1
000005                   10 WHERE      BSS      1
000006                   11 LENG       BSS      1
000007                   12 OP         BSS      1
000010                   13 MODE       BSS      1
000011 006017 A          14            LDAE     WHERE
000012 000005 R
000013 054047 A          15            STA      LRECL+1
000014 006017 A          16            LDAE*    LENG
000015 100006 R
000016 054043 A          17            STA      LRECL
000017 006017 A          18            LDAE*    MODE
000020 100010 R
000021 004244 A          19            LRLA     4
000022 006127 A          20            ADDE*    OP
000023 100007 R
000024 004250 A          21            LRLA     8
000025 006127 A          22            ADDE*    UNIT
000026 100004 R
000027 054003 A          23            STA      $+4
       000030 R          24 WRITE      EQU      $
                         25            WRITE    LRECL,5,,1   WILL BE MODIFIED
000030 006505 A
000031 000000 E
000032 100000 A
000033 010405 A
000034 000062 R
000035 000000 A
000036 000000 A
000037 006027 A          26            LDBE     WRITE+5
000040 000035 R
000041 010421 A          27            LDA      ONE
                         28            STAT     WRITE,ERR,ERR+1,ERR+2,ERR+3
000042 006505 A
000043 000000 E
000044 000030 R
000045 000053 R
000046 000054 R
000047 000055 R
000050 000056 R
000051 001000 A          29            RETU*    $$BUFF
000052 100000 R
000053 005311 A          30 ERR        DAR
000054 005311 A          31            DAR
```

```
                                   VORTEX  DASMR    $BUFF     0000 HOURS
000055 005311 A    32         DAR
000056 005311 A    33         DAR
000057 001000 A    34         RETU*    $$BUFF
000060 100000 R
000061             35 STA     BSS      1
                   36 LRECL   FCB      0,0,1
000062 000000 A
000063 000000 A
000064 000400 A
000065 000000 A
000066 000000 A
000067 000000 A
000070 000000 A
000071 000000 A
000072 000000 A
000073 000000 A
                   37         END
ENTRY NAMES
000000 R $$BUFF 000062 R LRECL
EXTERNAL NAMES
000002 E $SE    000031 E V$IOC  000043 E V$IOST
SYMBOLS
000000 R $$BUFF 000002 E $SE    000052 R ERR    000006 R LENG
000062 R LRECL  000010 R MODE   000421 A ONE    000007 R OP
000061 R STA    000004 R UNIT   000031 E V$IOC  000043 E V$IOST
000005 R WHERE  000030 R WRITE
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR   $BUFF     0000 HOURS

                        1          NAME     IUNPK
000000 006037 A         2 START    LDAE*    NTOWD
000001 100035 R
000002 006017 A         3          LDAE*    IPK
000003 100034 R
000004 134033 A         4          ADD      LLS
000005 054010 A         5          STA      LLS1
000006 014032 A         6          LDA      LASL
000007 006147 A         7          SUBE*    IPK
000010 100034 R
000011 054005 A         8          STA      LSL1
000012 006017 A         9          LDAE*    IA
000013 100032 R
000014 005344 A        10 DXR      DXR
000015 005002 A        11          TZB
000016 004540 A        12 LLS1     LLSR     0
000017 004117 A        13 LSL1     ASRB     15
000020 006067 A        14          STBE*    IB
000021 100033 R
000022 044010 A        15          INR      IB
000023 001046 A        16          JXNZ     DXR
000024 000014 R
000025 001000 A        17          JMP      *
000026 000025 R
000026                 18 IUNPK    BSS      0
                       19          EXT      $SE
000027 002000 A        20          CALL     $SE
000030 000000 E
000031 000004 A        21          DATA     4
000032 000000 A        22 IA       DATA     0
000033 000000 A        23 IB       DATA     0
000034 000000 A        24 IPK      DATA     0
000035 000000 A        25 NTOWD    DATA     0
000036 001000 A        26          JMP      START
000037 000000 R
000040 004540 A        27 LLS      LLSR     0
000041 004117 A        28 LASL     ASRB     15
                       29          END
ENTRY NAMES
000026 R IUNPK
EXTERNAL NAMES
000030 E $SE
SYMBOLS
000030 E $SE      000014 R DXR      000032 R IA       000033 R IB
000034 R IPK      000026 R IUNPK    000041 R LASL     000040 R LLS
000016 R LLS1     000017 R LSL1     000035 R NTOWD    000000 R START
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                   VORTEX  DASMR   $BUFF      0000 HOURS

                  1
                  2          NAME     ISHF
       000001 A   3 X        EQU      1
000000 006037 A   4 START    LDXE*    N
000001 100022 R
000002 005344 A   5 DXR      DXR
000003 006215 A   6          LDAE*    IA,X,0200
000004 100020 R
000005 004244 A   7          LRLA     4
000006 004304 A   8          ASRA     4
000007 006255 A   9          STAE*    IB,X,0200
000010 100021 R
000011 001046 A  10          JXNZ     DXR
000012 000002 R
000013 001000 A  11          JMP      *
000014 000013 R
000014           12 ISHF     BES      0
                 13          EXT      $SE
000015 002000 A  14          CALL     $SE
000016 000000 E
000017 000003 A  15          DATA     3
000020           16 IA       BSS      1
000021           17 IB       BSS      1
000022           18 N        BSS      1
000023 001000 A  19          JMP      START
000024 000000 R
                 20          END
ENTRY NAMES
000014 R ISHF
EXTERNAL NAMES
000016 E $SE
SYMBOLS
000016 E $SE     000002 R DXR     000020 R IA      000021 R IB
000014 R ISHF    000022 R N       000000 R START   000001 A X
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                     VORTEX  DASMR   $BUFF      0000 HOURS

                      1
                      2          NAME     $$BF1
                      3          EXT      $SE
000000 014044 A       4 START    LDA      FRST
000001 001016 A       5          JANZ     WRITE
000002 000025 R
000003 044041 A       6          INR      FRST
000004 005027 A       7          LDBE*    UNIT
000005 100041 R
000006 005021 A       8          TBA
000007 124011 A       9          ADD      REW+3
000010 054010 A      10          STA      REW+3
000011 005021 A      11          TBA
000012 124015 A      12          ADD      WRITE+3
000013 054014 A      13          STA      WRITE+3
000014 014025 A      14          LDA      IB
000015 054031 A      15          STA      FCB+1
                     16 REW      REW      FCB
000016 006505 A
000017 000000 E
000020 100000 A
000021 001400 A
000022 000046 R
000023 000000 A
000024 000000 A
                     17 WRITE    WRITE    FCB,,,3
000025 006505 A
000026 000017 E
000027 100000 A
000030 030400 A
000031 000046 R
000032 000000 A
000033 000000 A
000034 001000 A      18          JMP      *
000035 000034 R
000036               19 $$BF1    RES      0
000036 002000 A      20          CALL     $SE
000037 000000 E
000040 000002 A      21          DATA     2
000041 000000 A      22 UNIT     DATA     0
000042 000000 A      23 IB       DATA     0
000043 001000 A      24          JMP      START
000044 000000 R
000045 000000 A      25 FRST     DATA     0
                     26 FCB      FCB      245,0,1
000046 000365 A
000047 000000 A
000050 000400 A
000051 000000 A
000052 000000 A
000053 000000 A
000054 000000 A
000055 000000 A
```

```
                                          VORTEX  DASMR   $BUFF      0000 HOURS

000056 000000 A
000057 000000 A
                    27          END
ENTRY NAMES
000035 R $$BF1
EXTERNAL NAMES
000037 E $SE      000026 E V$IOC
SYMBOLS
000035 R $$BF1   000037 E $SE     000046 R FCB     000045 R FRST
000042 R IB      000016 R REW     000000 R START   000041 R UNIT
000026 E V$IOC   000025 R WRITE
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                          VORTEX  FTN IV             0000 HOURS

               SUBROUTINE DCMP(IUNIT,INUM,IA,NBUFF)
               DIMENSION IA(2)
               DATA NTMP,NN/0,1/
               INUM=0
        1     CALL IDCMP(IUNIT,INUM,NTMP,IA(NN),NN)
              NN=NN+IA(NN+2)
               IF(IA(NN).EQ.4095)RETURN
               GO TO 1
              END
ENTRY/COMMON BLOCK NAMES
  000110 R   DCMP
EXTERNAL NAMES
  000002 E     $SE
  000032 E   IDCMP
SYMBOL TABLE
  100004 R  IUNIT
  100005 R   INUM
  100006 R     IA
  100007 R  NBUFF
  000002 E    $SE
  000012 R   NTMP
  000013 R     NN
  000103 R 000000
  000021 R      1
  000032 E  IDCMP
  000104 R 177777
  000105 R    $1
  000027 R     0T
  000107 R 007777
  000106 R   $1 0
  0 ERRORS COMPILATION COMPLETE
```

```
PAGE    1                    VORTEX  FTN IV                    0000 HOURS

      SUBROUTINE IDCMP(IUNIT,INUM,NTMP,IA,NN)
      DIMENSION IA(1),IB(245)
      IF(IA(1).NE.1)GO TO 99
      INUM=INUM+1
       CALL ISHF(IA(4),IA(4),1)
       IST=IA(4)+1
       NWDS=240
      IF(IST.LT.0)GO TO 1
3      IF(NTMP.NE.0)CALL $$BF1(IUNIT,IB)
      NB=240
       IOUT=1
      NTMP=0
      IF(IST.GT.0)GO TO 4
      CALL ISHF(IA(2),IB,NWDS)
      GO TO 6
4     MM=16
      DO 11 I=1,3
      LL=50*I-49
11    CALL JDCMP(IA(I+4),IA(MM),IB(LL),50,MM)
      DO 12 I=1,3
      LL=10*I+141
12    CALL JDCMP(IA(I+7),IA(MM),IB(LL),10,MM)
      CALL ISHF(IA(MM),IB(181),9)
      MM=MM+9
      CALL JDCMP(IA(11),IA(MM),IB(190),23,MM)
      CALL JDCMP(IA(12),IA(MM),IB(213),19,MM)
      CALL ISHF(IA(MM),IB(232),13)
      NBEG=13
6     IB(NB+1)=IA(4)
      IB(NB+2)=IA(NBEG)
      IB(NB+3)=IA(NBEG+1)
      CALL ISHF(IA(NBEG+2),IB(NB+4),1)
      IB(NB+5)=NWDS
       IF(NTMP.NE.0)IB(NB)=-1
       IF(NTMP.EQ.48)IOUT=1
       IF(IOUT.EQ.0)RETURN
93     CALL$$BF1(IUNIT,IB)
       NTMP=0
       RETURN
99     IF(NTMP.EQ.0)RETURN
       IF(IA(NN+1).EQ.4095)GO TO 98
       RETURN
1      NWDS=IA(3)-7
       IF(NWDS.NE.0)GO TO 3
       IOUT=0
       NBEG=5
       NTMP=NTMP+1
       NB=245-5*NTMP
       GO TO 6
       END
ENTRY/COMMON BLOCK NAMES
```

```
  001447 R  IDCMP
EXTERNAL NAMES
  000002 E    $SE
  001131 E   ISHF
  001235 E  $$EF1
  001002 E  JDCMP
  000703 E    $DO
SYMBOL TABLE
  001367 R 000001
  001444 R 000002
  100004 R  IUNIT
  100005 R   INUM
  100006 R   NTMP
  100007 R     IA
  100010 R     NN
  000002 E    $SE
  000013 R     IS
  001446 R 000365
  001237 R     99
  001131 E   ISHF
  001365 R 000003
  001366 R     $1
  001356 R     03
  001370 R    IST
  001372 R   NUDS
  001371 R 000360
  001373 R 000000
  001306 R      1
  000467 R      3
  001235 E  $$EF1
  001374 R     NB
  001375 R   IOUT
  000552 R      4
  001376 R 000007
  001033 R      6
  001400 R     MN
  001377 R 000020
  000570 R     11
  001401 R      I
  001404 R     LL
  001402 R 000002
  001403 R 000001
  001002 E  JDCMP
  001405 R 177777
  001406 R   $1 0
  001407 R 000012
  001410 R   $1 1
  000703 E    $DO
  000646 R     12
  001411 R 000013
  001412 R 000012
  001413 R 000002
  001415 R 000011
```

```
001414 R 000277
001420 R 000013
001417 R 000027
001416 R 000310
001422 R 000017
001421 R 000337
001424 R 000015
001423 R 000356
001425 R   NBEG
001426 R 000013
001427 R 000014
001430 R    $1 2
001431 R    $1 3
001432 R    $1 4
001433 R 000015
001434 R    $1 5
001435 R    $1 6
001436 R    $1 7
001437 R 000016
001440 R    $1 8
001441 R 000017
001442 R 000060
001224 R      98
001443 R 007777
001445 R 000005
0 ERRORS COMPILATION COMPLETE
```

```
          SUBROUTINE JDCMP(IPK,IA,IB,N,MM)
          DIMENSION IB(1),IA(2)
          IF(IPK.NE.12)GO TO 1
          CALL ISHF(IA,IB,N)
          MM=MM+N
          RETURN
    1     IF(IPK.NE.0)GO TO 2
          CALL ISHF(IA,IB,1)
          DO 22 I=2,N
   22     IB(I)=IB(1)
          MM=MM+1
          RETURN
    2     CALL ISHF(IA,IB,1)
          NTOWD=12/IPK
          L=2
          NN=N-1
          DO 3 I=1,NN,NTOWD
          CALL IUNPK(IA(L),IB(I+1),IPK,NTOWD)
          L=L+1
    3     CONTINUE
          MM=MM+L
          DO 4 I=2,N
    4     IB(I)=IB(I)+IB(I-1)
          RETURN
          END
ENTRY/COMMON BLOCK NAMES
  000341 R  JDCMP
EXTERNAL NAMES
  000002 E     $SE
  000145 E    ISHF
  000315 E     $DO
  000225 E   IUNPK
SYMBOL TABLE
  000326 R 000001
  000327 R 000002
  100004 R     IPK
  100005 R      IA
  100006 R      IB
  100007 R       N
  100010 R      MM
  000002 E     $SE
  000324 R 000014
  000311 R      09
  000054 R       1
  000145 E    ISHF
  000325 R 000000
  000144 R       2
  000101 R      22
  000330 R       I
  000331 R 177777
  000332 R   $1
  000315 E     $DO
  000333 R   NTOWD
```



```
  000334 R       L
  000335 R      NN
  000237 R       3
  000225 E   IUNPK
  000336 R   $1 0
  000264 R       4
  000337 R 177776
  000340 R   $1 1
0 ERRORS COMPILATION COMPLETE
```

Appendix C

Each physical record consists of 2560 or fewer 12-bit words, (512 CDC 6600 words), and contains as many logical records as can be completely contained in 2559 words. All logical records contain 3 control words, except for type -1.

Word 1 = type of record
Word 2 = subtype
Word 3 = number of 12-bit words this logical record (including control words).

At present, three types of records have been reserved.

Type 0 = header record, not yet defined
Type 1 = frame data
Type -1 = end of physical record (12-bit, 2's complement -1).

For type 1 records, subtype 0 has been defined.

Word 1 = 1
Word 2 = 0
Word 3 = N = number of words
Word 4 = Status = 1 - good frame; -1 - error frame; -2 - timeout

The rest of the frame is as follows for the various frame statuses.

	Status		
	-2	-1	-1
Word 5	"received"	time	seconds
Word 6	"	"	milliseconds
Word 7	Buffer contents		
Word 8-N	partial frame received, if any.		240 data words received in arranged form by instrument as given in AFCRL-TR-75-0588.

Data and buffer contents are two's complement

11	10 … 1	0
s	data	e

Status 1

Word 5 - packing factor SCX
Word 6 - " " SCY
Word 7 - " " SCZ
Word 8 - " " FFX
Word 9 - " " FFY
Word 10 - " " FFZ

Word 11 - packing factor Analog spares
Word 12 - " " Digital spares

The rest of the frame is then the same as words 5-N of status -1 frames except that relevant data is compressed according to the scheme detailed below.

Packing Scheme

If a particular instrument gives in readings, $X_1, \ldots\ldots, X_m$, m-1 ΔX_i are calculated where $\Delta X_i = X_{i+1} - X_i$. If all the ΔX_i are 0, the packing factor K is chosen to be 0. Otherwise K is the minimum of 2, 3, 4, 6, 12 such that $-2^{K-1} \leq \Delta X_i \leq 2^{K-1} -1$ for all i=1,...,m=1.

When K=12, actual data is stored. When K≠12, the packed instrument format is as follows, where $\ell = 12/K$

X_1		
ΔX_ℓ	...	ΔX_1
ΔX_ℓ	...	$\Delta X_{\ell+1}$
	⋮	
	ΔX_{m-1}	...
X_m		

for K ≠ 0

It should be noted that X_1 and X_m are stored with error bits (which are zero because we only pack error-free frames) but that the ΔX_i are data deltas only, in two's complement form.

Therefore, after the ΔX_i's are expanded to full words, $X_i = X_{i-1} + 2\Delta X_{i-1}$.

2. UNPACKING AND PACKING SUBROUTINES

SUBROUTINES $UNPK AND $REPK

SUBROUTINES $UNPK AND $REPK HAVE BEEN WRITTEN RESPECTIVELY TO UNPACK A DATA FRAME FROM RECEIVED DATA ORDER TO INSTRUMENT ORDER (DEFINED IN A PREVIOUS REPORT) AND TO REPACK THE DATA INTO RECEIVED DATA ORDER. SUBROUTINE REPACK IS USED PRIMARILY TO OBTAIN DUMPS FOR HARDWARE DIAGNOSTIC PURPOSES.

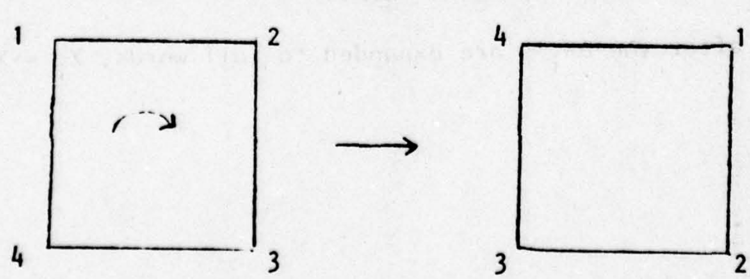
A PERMUTATION TABLE ($CYCST THRU Ø$CYCND) WHICH DEFINES THE PARTICULAR PERMUTATION FOR THIS PURPOSE HAS BEEN ADDED TO THE IN CORE SYSTEM.

```
USAGE:  DIMENSIØN IA(245), IB(245)
        CALL $UNPK(IA,IB)
        CALL $REPK(IA,IB)
RESTRICTION:  IA AND IB MUST BE DISJOINT OR
              IDENTICAL.
```

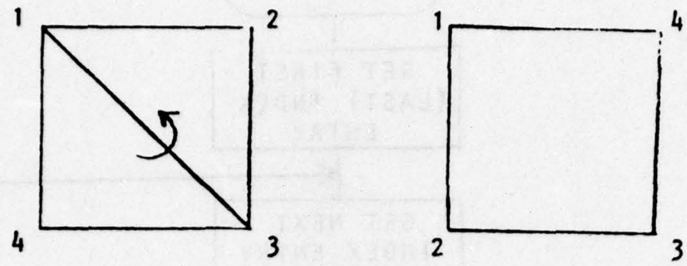
BECAUSE THE METHOD USED IS GENERALLY APPLICABLE TO ANY PERMUTATION AND ITS INVERSE, A BRIEF DISCUSSION OF METHOD AND A FLOW CHART OF THE PROCEDURE WILL BE GIVEN.

THEORY:

ANY PERMUTATION CAN BE REPRESENTED AS A PRODUCT OF DISJOINT CYCLES, WHERE A CYCLE IS MERELY A TRACING OF ELEMENTS WHICH PERMUTE INTO EACH OTHER. (SEE ANY TEXT ON MODERN ALGEBRA). FOR EXAMPLE, USING PERMUTATIONS OF A SQUARE, ROTATION THROUGH 90°:



CAN BE REPRESENTED AS (1,2,3,4) i.e. 1 GOES TO 2 GOES TO 3 GOES TO 4 GOES TO 1 AND REFLECTION ABOUT A DIAGONAL



CAN BE REPRESENTED AS (1) (3) (2,4).

THE ADVANTAGE OF THIS REPRESENTATION IS THAT THE INVERSE PERMUTATION IS IMPLICITLY DEFINED BY THE PERMUTATION ITSELF - READ THE CYCLES BACKWARDS.

METHOD:

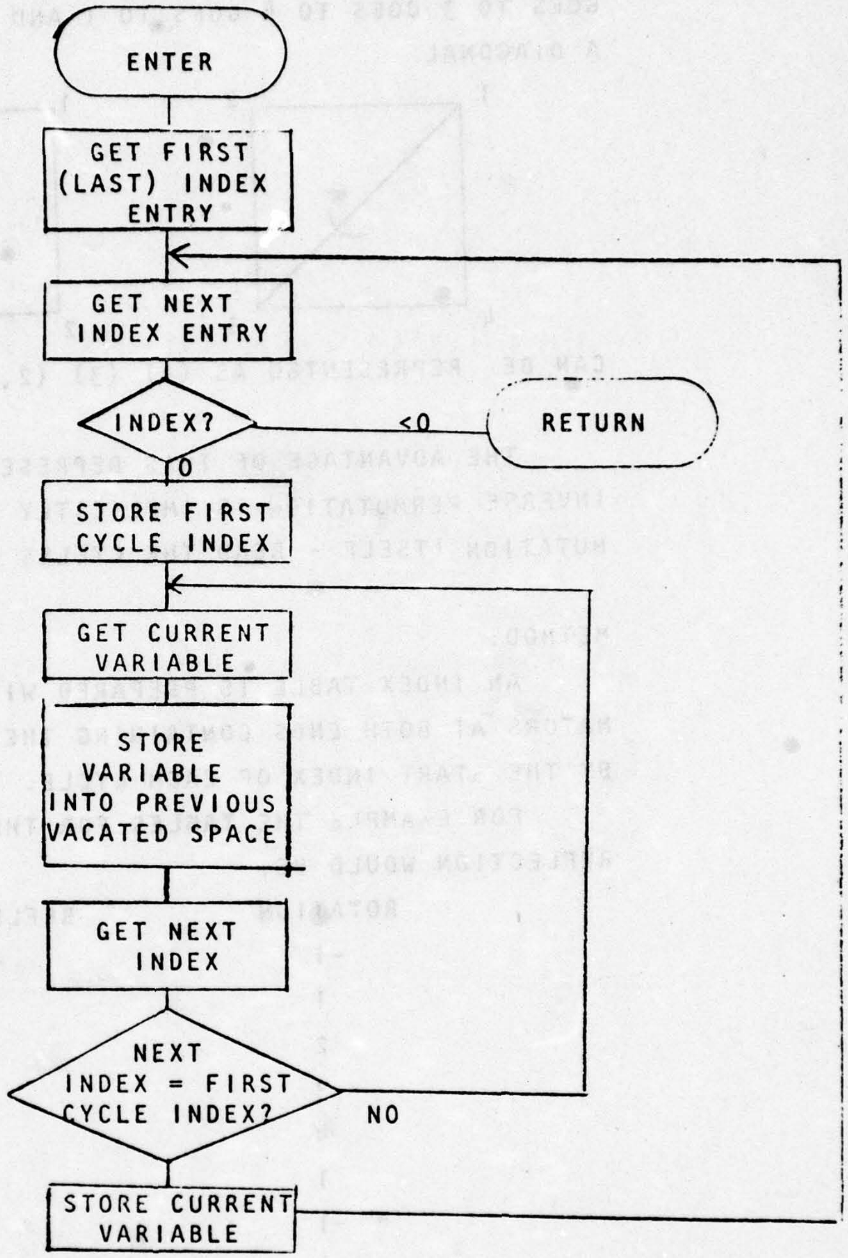
AN INDEX TABLE IS PREPARED WITH NEGATIVE TERMINATORS AT BOTH ENDS CONTAINING THE CYCLES AUGMENTED BY THE START INDEX OF EACH CYCLE.

FOR EXAMPLE THE TABLES FOR THE ROTATION AND REFLECTION WOULD BE:

ROTATION	REFLECTION
-1	-1
1	1
2	1
3	3
4	3
1	2
-1	4
	2
	-1

THE PROGRAMS THEN SCAN THE CYCLES IN THE INDEX TABLE IN THE PROPER DIRECTION TO PERFORM THE PERMUTATION.

FLOW CHART:



ENTER
GET FIRST (LAST) INDEX ENTRY
GET NEXT INDEX ENTRY
INDEX?
<0
RETURN
0
STORE FIRST CYCLE INDEX
GET CURRENT VARIABLE
STORE VARIABLE INTO PREVIOUS VACATED SPACE
GET NEXT INDEX
NEXT INDEX = FIRST CYCLE INDEX?
NO
STORE CURRENT VARIABLE

```
        000001 A    1 X       EQU     1
        000002 A    2 B       EQU     2
000000 014042 A    3 SETUP   LDA     AA
000001 054011 A    4         STA     AGIN+1
000002 014041 A    5         LDA     AA+1
000003 054024 A    6         STA     BB1+1
000004 054010 A    7         STA     BB2+1
                   8         EXT     $CYCST,$CYCND
000005 006030 A    9         LDXI    $CYCND
000006 000000 E                                         ($CYCST FOR $REPK
000007 001000 A   10         JMP     DXR1
000010 000031 R
000011 064036 A   11 AGIN1   STB     TEMP1
000012 006030 A   12 AGIN    LDXE    AA,B
000013 000043 R
000014 006056 A   13 BB2     STAE    BB,B
000015 000044 R
000016 005041 A   14         TXA
000017 034027 A   15         LDX     TEMP          (IXR FOR $REPK)
000020 005344 A   16         DXR
000021 074025 A   17         STX     TEMP
000022 025000 A   18         LDB     0,X
000023 006627 A   19         SBE     TEMP1,7,020
000024 000050 R
000025 001000 A   20         JMP     AGIN
000026 000012 R
000027 006056 A   21 BB1     STAE    BB,B
000030 000044 R                                         (IXR FOR $REPK)
000031 005344 A   22 DXR1    DXR
000032 074014 A   23         STX     TEMP
000033 025000 A   24         LDB     0,X
000034 006477 A   25         BT      077,AGIN1       BTD0
000035 000011 R
000036 001000 A   26         JMP     *
000037 000036 R                                         ($REPK FOR $REPK)
                  27         NAME    $UNPK
                  28         EXT     $SE                ($REPK FOR $REPK
000037            29 $UNPK   BES     0
000040 002000 A   30         CALL    $SE,2
000041 000000 E
000042 000002 A
000043 000000 A   31 AA      DATA    0,0
000044 000000 A
000044            32 BB      BES     0
000045 001000 A   33         JMP     SETUP
000046 000000 R
000047 000000 A   34 TEMP    DATA    0
000050 000000 A   35 TEMP1   DATA    0
                  36         END
ENTRY NAMES
000037 R $UNPK
EXTERNAL NAMES
000006 E $CYCND 000000 E $CYCST 000041 E $SE

SYMBOLS
000006 E $CYCND 000000 E $CYCST 000041 E $SE     000037 R $UNPK
000043 R AA     000012 R AGIN   000011 R AGIN1  000002 A B
000044 R BB     000027 R BB1    000014 R BB2    000031 R DXR1
000000 R SETUP  000047 R TEMP   000050 R TEMP1  000001 A X
  0 ERRORS ASSEMBLY COMPLETE
```

```
                         1
                         2
                         3          NAME     $CYCST,$CYCND
000000 177777 A          4 $CYCST   DATA     -1
                         5 *        NLIS
000001 000000 A          6          DATA     0,4,16,64,55,24,92,179,189,3,13,51,8,32,140
000002 000004 A
000003 000020 A
000004 000100 A
000005 000067 A
000006 000030 A
000007 000134 A
000010 000263 A
000011 000275 A
000012 000003 A
000013 000015 A
000014 000063 A
000015 000010 A
000016 000040 A
000017 000214 A
000020 000256 A          7          DATA     174,78,125,117,69,74,93,182,212,215,218,221,224,227,95
000021 000116 A
000022 000175 A
000023 000165 A
000024 000105 A
000025 000112 A
000026 000135 A
000027 000266 A
000030 000324 A
000031 000327 A
000032 000332 A
000033 000335 A
000034 000340 A
000035 000343 A
000036 000137 A
000037 000300 A          8          DATA     192,60,43,181,209,108,34,146,196,152,38,162,39,165,112
000040 000074 A
000041 000053 A
000042 000265 A
000043 000321 A
000044 000154 A
000045 000042 A
000046 000222 A
000047 000304 A
000050 000230 A
000051 000046 A
000052 000242 A
000053 000047 A
000054 000245 A
000055 000160 A
000056 000062 A          9          DATA     50,5,23,89,166,131,139,167,150,0
000057 000005 A
000060 000027 A
```

```
000061 000131 A
000062 000246 A
000063 000203 A
000064 000213 A
000065 000247 A
000066 000226 A
000067 000000 A
000070 000001 A   10       DATA    1,7,29,127,123,91,176,132,142,180,206,105,25,115,63
000071 000007 A
000072 000035 A
000073 000177 A
000074 000173 A
000075 000133 A
000076 000260 A
000077 000204 A
000100 000216 A
000101 000264 A
000102 000316 A
000103 000151 A
000104 000031 A
000105 000163 A
000106 000077 A
000107 000064 A   11       DATA    52,11,45,191,41,175,113,53,14,54,17,67,68,71,84
000110 000013 A
000111 000055 A
000112 000277 A
000113 000051 A
000114 000257 A
000115 000161 A
000116 000065 A
000117 000016 A
000120 000066 A
000121 000021 A
000122 000103 A
000123 000104 A
000124 000107 A
000125 000124 A
000126 000223 A   12       DATA    147,199,98,201,100,6,26,118,72,87,160,1
000127 000307 A
000130 000142 A
000131 000311 A
000132 000144 A
000133 000006 A
000134 000032 A
000135 000166 A
000136 000110 A
000137 000127 A
000140 000240 A
000141 000001 A
000142 000002 A   13       DATA    2,10,42,178,170,2
000143 000012 A
000144 000052 A
000145 000262 A
```

```
000146 000252 A
000147 000002 A
000150 000011 A    14        DATA    9,35,153,57,30,134,143,202,101,9
000151 000043 A
000152 000231 A
000153 000071 A
000154 000036 A
000155 000206 A
000156 000224 A
000157 000312 A
000160 000145 A
000161 000011 A
000162 000014 A    15        DATA    12,48,200,99,204,103,15,61,46,194,114,56,27,121,85
000163 000060 A
000164 000310 A
000165 000143 A
000166 000314 A
000167 000147 A
000170 000017 A
000171 000075 A
000172 000056 A
000173 000302 A
000174 000162 A
000175 000070 A
000176 000033 A
000177 000171 A
000200 000125 A
000201 000232 A    16        DATA    154,76,119,75,116,66,65,62,49,203,102,12
000202 000114 A
000203 000167 A
000204 000113 A
000205 000164 A
000206 000102 A
000207 000101 A
000210 000076 A
000211 000061 A
000212 000313 A
000213 000146 A
000214 000014 A
000215 000022 A    17        DATA    18,70,81,138,164,77,122,88,163,58,33,143,183,207,106
000216 000106 A
000217 000121 A
000220 000212 A
000221 000244 A
000222 000115 A
000223 000172 A
000224 000130 A
000225 000243 A
000226 000072 A
000227 000041 A
000230 000217 A
000231 000267 A
000232 000317 A
```

```
000233 000152 A
000234 000034 A    18        DATA     28,124,94,185,213,216,219,222,225,228,96,195,133,145,193
000235 000174 A
000236 000136 A
000237 000271 A
000240 000325 A
000241 000330 A
000242 000333 A
000243 000336 A
000244 000341 A
000245 000344 A
000246 000140 A
000247 000303 A
000250 000205 A
000251 000221 A
000252 000301 A
000253 000117 A    19        DATA     79,128,126,120,82,141,177,151,19,73,90,173,59,36,156
000254 000200 A
000255 000176 A
000256 000170 A
000257 000122 A
000260 000215 A
000261 000261 A
000262 000227 A
000263 000023 A
000264 000111 A
000265 000132 A
000266 000255 A
000267 000073 A
000270 000044 A
000271 000234 A
000272 000202 A    20        DATA     130,136,158,168,169,188,214,217,220,223,226,229,97,198,19
000273 000210 A
000274 000236 A
000275 000250 A
000276 000251 A
000277 000274 A
000300 000326 A
000301 000331 A
000302 000334 A
000303 000337 A
000304 000342 A
000305 000345 A
000306 000141 A
000307 000306 A
000310 000276 A
000311 000026 A    21        DATA     22,86,157,149,205,104,18
000312 000126 A
000313 000235 A
000314 000225 A
000315 000315 A
000316 000150 A
000317 000022 A
```

```
000320 000024 A   22          DATA     20,80,135,155,111,47,197,171,21,83,144,186,208,107,31
000321 000120 A
000322 000207 A
000323 000233 A
000324 000157 A
000325 000057 A
000326 000305 A
000327 000253 A
000330 000025 A
000331 000123 A
000332 000220 A
000333 000272 A
000334 000320 A
000335 000153 A
000336 000037 A
000337 000211 A   23          DATA     137,161,20
000340 000241 A
000341 000024 A
000342 000045 A   24          DATA     37,159,187,211,110,44,184,210,109,37
000343 000237 A
000344 000273 A
000345 000323 A
000346 000156 A
000347 000054 A
000350 000270 A
000351 000322 A
000352 000155 A
000353 000045 A
000354 000050 A   25          DATA     40,172,40
000355 000254 A
000356 000050 A
000357 000201 A   26          DATA     129,129
000360 000201 A
000361 000346 A   27          DATA     230,230
000362 000346 A
000363 000347 A   28          DATA     231,231
000364 000347 A
000365 000350 A   29          DATA     232,232
000366 000350 A
000367 000351 A   30          DATA     233,233
000370 000351 A
000371 000352 A   31          DATA     234,234
000372 000352 A
000373 000353 A   32          DATA     235,235
000374 000353 A
000375 000354 A   33          DATA     236,236
000376 000354 A
000377 000355 A   34          DATA     237,237
000400 000355 A
000401 000356 A   35          DATA     238,238
000402 000356 A
000403 000357 A   36          DATA     239,239
000404 000357 A
000405 000360 A   37          DATA     240,240
000406 000360 A
000407 000361 A   38          DATA     241,241
000410 000361 A
000411 000362 A   39          DATA     242,242
000412 000362 A
000413 000363 A   40          DATA     243,243
000414 000363 A
000415 000364 A   41          DATA     244,244
000416 000364 A
                  42          LIST
000417 177777 A   43 $CYCND   DATA     -1
                  44          END
ENTRY NAMES
000417 R $CYCND 000000 R $CYCST
EXTERNAL NAMES
SYMBOLS
000417 R $CYCND 000000 R $CYCST
  0 ERRORS ASSEMBLY COMPLETE
```

3. DATA STORAGE AND RETRIEVAL SUBROUTINES

THE DATA STORAGE AND RETRIEVAL SUBROUTINES ARE DESIGNED TO FACILITATE THE STORAGE AND RETRIEVAL OF TIME SERIES DATA WITHOUT REQUIRING INORDINATE AMOUNTS OF CORE. THERE ARE SIX SUBROUTINES, EACH OF WHICH WILL BE DESCRIBED.

ROUTINE: OP$N

CALLING SEQUENCE: CALL OP$N (BUFFER,NVAR)

- BUFFER - FLOATING POINT ARRAY OF DIMENSION AT LEAST 60*NVAR (120*NVAR FOR FIXED POINT)
- NVAR - NUMBER OF VARIABLES
- OP$N - MUST BE CALLED BEFORE ANY OTHER ROUTINES TO OPEN THE DISKFILE (PLOTFL ON UNTIL 30) AND SETUP FOR OTHER ROUTINES

ROUTINE: ADDCMP (ADD COMPONENT)

CALLING SEQUENCE: CALL ADDCMP(A,I)

- A - VARIABLE TO BE ADDED
- I - INDEX OF VARIABLE SNNAR

ROUTINE: RETRIV

CALLING SEQUENCE: CALL RETRIV(VECT,I)

- VECT - VECTOR WHICH IS TO BE RETURNED
- I - INDEX OF VARIABLE

ROUTINE: CLO$E

CALLING SEQUENCE: CALL CLO$E

CLOSES DISKFILE AND UPDATES IT. REOPENS IT FOR LATER USE

<u>NOTE</u>: FOR PROPER PROGRAM FUNCTIONING, ADDCMP MUST HAVE BEEN CALLED THE SAME NUMBER OF TIMES FOR EACH INDEX, AND THAT NUMBER MUST BE A MULTIPLE OF 60.

ROUTINE: CLO$E

CALLING SEQUENCE: CALL CLO$E

CLOSE DISK FILE AND UP DATE IT. REOPEN IT FOR LATER USE.

NOTE: FOR PROPER PROGRAM FUNCTIONING, ADDCMP MUST HAVE BEEN CALLED THE SAME NUMBER OF TIMES FOR EACH INDEX AND THAT NUMBER MUST BE A MULTIPLE OF 60.

ROUTINE: OP$N1, CLO$1

CALLING SEQUENCE: CALL (OP$N1) (IFIRST, ILAST)
(CLO$S1)

(OP$N1 RETRIEVES) (CLO$1 STORES) DATA BETWEEN IFIRST AND ILAST (FROM) (TO) DISK.

EXAMPLE: A TAPE EXISTS IN THE FOLLOWING CARD IMAGE FORMAT.

RECORD 1 - Alphabetic ID.

RECORD 2 - n_1 - X_1 - X_{20} - OBSERVATIONS OF 20 VARIABLES (20F4.0) $n \leq 2401$

PROBLEM: TO STORE THE DATA MATRIX ON DISK, LATER TO PRINT OUT THE DATA VARIABLE BY VARIABLE.

```
      DIMENSION ID(40) BUFFER(1200) X(20)
      COMMON /XXXX/ ID.N
      REWIND 21

      READ (21.100) ID
  100 FORMAT (40A2)
      N=0

      CALL OP$N(BUFFER.20)
    1 READ (21.101) X
  101 FORMAT (20F40)

      IF(IOCHK(21))2.3.4
    2 BACKSPACE 21
      GO TO 1

    3 N=N+1
      DO 5 I=1. 20
    5 CALL ADDCMP(X(I).I)

      GO TO 1
    4 NADD=MOD(N.60)
      IF(NADD. EQ. 0) GO TO 99

 DO 6 I=1.NADD
      DO 6 J=1.20
    6 CALL ADDCMP(0.0.J)

   99 CALL CLOS1(ID.ID(41))
      STOP
      END
      DIMENSION ID(41). VECT(2400)
      CALL OP$N(VECT.20)
      CALL OP$N1(ID.ID(41))

      N=ID(41)
      DO 1 I=1.20
      CALL RETRIV (VECT.I)

    1 WRITE (5.100) (ID(J).J=1.40). (VECT(J)2
    1 WRITE (5.100) (ID(J). J=1.40). (VECT(J)J=1.N)
  100 FORMAT(1H1.40A2/(1X.30F4.0))

      END
```

```
                     1          NAME     OP$N,CLO$E,RETRIV,ADDCMP,OP$N
                     2          EXT      $SE                      CLOSE1
       000422 A      3 TWO      EQU      0422
       000001 A      4 X        EQU      1
       000002 A      5 B        EQU      2
000000 000000 A      6 OP$N1    ENTR
000001 002000 A      7          CALL     $SE,2,0,0
000002 000000 E
000003 000002 A
000004 000000 A
000005 000000 A
000006 002000 A      8          CALL     SAVE
000007 000062 R
000010 005001 A      9          TZA
000011 006506 A     10          JSR      RDW,B
000012 000026 R
000013 000000 A     11 CLO$1    ENTR
000014 002000 A     12          CALL     $SE,2,0,0
000015 000002 E
000016 000002 A
000017 000000 A
000020 000000 A
000021 002000 A     13          CALL     SAVE
000022 000062 R
000023 005101 A     14          INCR     01
000024 006506 A     15          JSR      RDW,B
000025 000026 R
000026 004250 A     16 RDW      LPLA     S
000027 124073 A     17          ADD      READ+3
000030 054021 A     18          STA      RDW1+3
000031 005101 A     19          INCR     01
000032 054304 A     20          STA      FCB+3
000033 005021 A     21          TBA
000034 006140 A     22          SUBI     CLO$1-OP$N1
000035 000013 A
000036 005014 A     23          TAX
000037 015000 A     24          LDA      0,X
000040 054027 A     25          STA      RETRN
000041 015004 A     26          LDA      4,X
000042 054272 A     27          STA      FCB+1
000043 005211 A     28          CPA
000044 120422 A     29          ADD      TWO
000045 125005 A     30          ADD      5,X
000046 054265 A     31          STA      FCB
                    32 RDW1     READ     FCB,30
000047 006505 A
000050 000000 E
000051 100000 A
000052 000036 A
000053 000334 R
000054 000000 A
000055 000000 A
000056 014267 A     33          LDA      D120
```

```
000057 054254 A    34          STA     FCB
000060 001000 A    35          JMP     RETRN+1
000061 000071 R
000062 000000 A    36 SAVE     ENTR
000063 054012 A    37          STA     STA
000064 064012 A    38          STB     STB
000065 074012 A    39          STX     STX
000066 001000 A    40          RETU*   SAVE
000067 100062 R
000070 000000 A    41 RETRN    ENTR
000071 014004 A    42          LDA     STA
000072 024004 A    43          LDB     STB
000073 034004 A    44          LDX     STX
000074 001000 A    45          RETU*   RETRN
000075 100070 R
000076 000000 A    46 STA      DATA    0
000077 000000 A    47 STB      DATA    0,0
000100 000000 A
000100             48 STX      BES     0
                   49 *
                   50 *
000101 002000 A    51 RETR1    CALL    SAVE
000102 000062 R
000103 014045 A    52          LDA     WHERE
000104 054230 A    53          STA     FCB+1
                   54          OPEN    FCB,30
000105 006505 A
000106 000050 E
000107 100000 A
000110 003036 A
000111 000334 R
000112 000000 A
000113 000000 A
000114 006017 A    55          LDAE*   ICM1
000115 100152 R
000116 005111 A    56          IAR
000117 054217 A    57          STA     FCB+3
                   58 READ     READ    FCB,30
000120 006505 A
000121 000106 E
000122 100000 A
000123 000036 A
000124 000334 R
000125 000000 A
000126 000000 A
000127 014205 A    59          LDA     FCB+1
000130 124215 A    60          ADD     D120
000131 054203 A    61          STA     FCB+1
000132 014204 A    62          LDA     FCB+3
000133 124175 A    63          ADD     NCONP
000134 054202 A    64          STA     FCB+3
000135 124203 A    65          ADD     FCB+5
000136 144201 A    66          SUB     FCB+4
```

```
000137 005311 A   67            DAR
000140 001004 A   68            JAN       READ
000141 000120 R
000142 002000 A   69            CALL      RETRN
000143 000070 R
000144 001000 A   70            JMP       *
000145 000144 R
000145            71 RETRIV     BES       0
000146 002000 A   72            CALL      $SE,2
000147 000015 E
000150 000002 A
000151 000000 A   73 WHERE      DATA      0
000152 000000 A   74 ICM1       DATA      0
000153 001000 A   75            JMP       RETR1
000154 000101 R
                  76 *
                  77 *
000155 002000 A   78 ADDCM1     CALL      SAVE
000156 000062 R
000157 006027 A   79            LDBE*     ICOMP
000160 100220 R
000161 005322 A   80            DBR
000162 064035 A   81            STB       ICOMP
000163 006216 A   82            LDAE      STRCMP,B,0200
000164 000427 R
000165 054147 A   83            STA       FCB+1
000166 006326 A   84            ADDE      NCMPS,B,0200
000167 000377 R
000170 005014 A   85            TAX
000171 024025 A   86            LDB       A
000172 016000 A   87            LDA       0,B
000173 055000 A   88            STA       0,X
000174 016001 A   89            LDA       1,B
000175 055001 A   90            STA       1,X
000176 024021 A   91            LDB       ICOMP
000177 006216 A   92            LDAE      NCMPS,B,0200
000200 000377 R
000201 120422 A   93            ADD       TWO
000202 005014 A   94            TAX
000203 144142 A   95            SUB       D120
000204 001010 A   96            JAZ       WRITE
000205 000223 R
000206 006276 A   97 RET1       STXE      NCMPS,B,0200
000207 000377 R
000210 002000 A   98            CALL      RETRN
000211 000070 R
000212 001000 A   99            JMP       *
000213 000212 R
000213           100 ADDCMP     BES       0
000214 002000 A  101            CALL      $SE,2
000215 000147 E
000216 000002 A
000217 000000 A  102 A          DATA      0
```

```
000220 000000 A   103 ICOMP  DATA    0
000221 001000 A   104        JMP     ADDCM1
000222 000155 R
000223 006216 A   105 WRITE  LDAE    RECNO,B,0200
000224 000347 R
000225 054111 A   106        STA     FCB+3
000226 124102 A   107        ADD     NCOMP
000227 006256 A   108        STAE    RECNO,B,0200
000230 000347 R
                  109        WRITE   FCB,30
000231 006505 A
000232 000121 E
000233 100000 A
000234 000436 A
000235 000334 R
000236 000000 A
000237 000000 A
000240 005004 A   110        TZX
000241 001000 A   111        JMP     RET1
000242 000206 R
                  112 *
                  113 *
000243 000000 A   114 CLO$E  ENTR
000244 002000 A   115        CALL    SAVE
000245 000062 R
000246 014100 A   116        LDA     RECNO
000247 054067 A   117        STA     FCB+3
                  118        CLOSE   FCB,30,,1
000250 006505 A
000251 000232 E
000252 100000 A
000253 013436 A
000254 000334 R
000255 000000 A
000256 000000 A
000257 002000 A   119        CALL    RETRN
000260 000070 R
000261 001000 A   120        RETU*   CLO$E
000262 100243 R
                  121 *
                  122 *
000263 002000 A   123 OPN1   CALL    SAVE
000264 000062 R
000265 006037 A   124        LDXE*   NCOMP
000266 100331 R
000267 074041 A   125        STX     NCOMP
000270 005042 A   126        TXB
000271 014036 A   127        LDA     WORK
000272 164053 A   128        MUL     D120
000273 005021 A   129        TBA
000274 005021 A   130        TBA
000275 005002 A   131        TZB
000276 144047 A   132 NXT1   SUB     D120
```

```
000277 006255 A  133            STAE     STRCMP-1,X,0200
000300 000426 R
000301 006275 A  134            STXE     RECNO-1,X,0200
000302 000346 R
000303 006245 A  135            INRE     RECNO-1,X,0200
000304 000346 R
000305 006265 A  136            STDE     NCMPS-1,X,0200
000306 000376 R
000307 005344 A  137            DXR
000310 001046 A  138            JXNZ     NXT1
000311 000276 R
                 139            OPEN     FCB,30
000312 006505 A
000313 000251 E
000314 100000 A
000315 003036 A
000316 000334 R
000317 000000 A
000320 000000 A
000321 002000 A  140            CALL     RETRN
000322 000070 R
000323 001000 A  141            JMP      *
000324 000323 R
000324           142 OP$N       BES      0
000325 002000 A  143            CALL     $SE,2
000326 000215 E
000327 000002 A
000330 000000 A  144 WORK       DATA     0
000331 000000 A  145 NCOMP      DATA     0
000332 001000 A  146            JMP      OPN1
000333 000263 R
                 147 FCB        FCB      120,*,0,,'PL','OT','FL'
000334 000170 A
000335 000334 R
000336 000000 A
000337 000000 A
000340 000000 A
000341 000000 A
000342 000000 A
000343 150314 A
000344 147724 A
000345 143314 A
000346 000170 A  148 D120       DATA     120
000347           149 RECNO      BSS      24
000377           150 NCMPS      BSS      24
000427           151 STRCMP     BSS      24
                 152            END
ENTRY NAMES
000213 R ADDCMP 000013 R CLO#1  000243 R CLO$E  000324 R OP$N
000000 R OP#N1  000145 R RETRIV
EXTERNAL NAMES
000326 E $SE    000313 E V$IOC
SYMBOLS
```

```
000326 E $SE     000217 R A        000155 R ADDCM1 000213 R ADDCMP
000002 A B       000013 P CLO#1    000243 P CLO#E  000346 R D120
000334 R FCB     000152 R ICM1     000220 R ICOMP  000377 R NCMPS
000331 R NCOMP   000276 R NXT1     000324 P OP#N   000000 R OP#N1
000263 R OPN1    000026 R RDW      000047 R RDW1   000120 R READ
000347 R RECNO   000206 R RET1     000101 R RETR1  000145 R RETRIV
000070 R RETRN   000062 R SAVE     000076 R STA    000077 R STB
000427 R STRCMP  000100 R STX      000422 A TWO    000313 E V$IOC
000151 R WHERE   000330 R WORK     000223 R WRITE  000001 A X
  0 ERRORS ASSEMBLY COMPLETE
```

4. TEKTRONIX PLOTTING ROUTINES

A SERIES OF PLOTTING ROUTINES HAS BEEN ADDED TO THE SYSTEM TO ENABLE PLOT FILES IN VARIAN DATAPLOT FORMAT TO BE OUTPUT TO THE TEKTRONIX 4014. THESE ROUTINES SCALE A STATOS PLOT SO THAT ONE INCH ON THE STATOS EQUALS ONE INCH ON CRT, NOT ON THE HARDCOPY. THESE ROUTINES ARE TRANSPARENT TO THE USER, WITH THE FOLLOWING EXCEPTIONS:

1) CARE SHOULD BE USED WITH NEGATIVE ORIGINS (in inches)

2) PRINTS BEYOND THE SCREEN LIMITS END UP AT THE RIGHT HAND AND TOP LIMITS OF THE SCREEN

3) ALL CHARACTERS ARE PLOTTED AT THE CURRENT CHARACTER SIZE (see TEKFNC below) AND UPRIGHT-ORIENTATION

4) SPECIAL CHARACTERS HAVE NOT BEEN IMPLEMENTED

5) MINIMIZATION OF STATOS SORT AND PLOT TIMES MAY INCREASE TEKTRONIX PLOT TIMES

IN ADDITION, A SUBROUTINE TEKFNC HAS BEEN WRITTEN TO ENABLE THE USER ACCESS TO TEKTRONIX FUNCTIONS.

USAGE:

TO OUTPUT TO THE TEKTRONIX, ADD THE FOLLOWING SUBROUTINES TO ANY DATAPLOT PROGRAM:

```
SUBROUTINE DPSORT
CALL CRTPLT
RETURN
END
SUBROUTINE DPPLOT
RETURN
END
```

TO USE TEKTRONIX FUNCTIONS:

```
CALL TEKFNC(I)

I------------------------------FUNCTION
1------------------------------LARGEST CHARACTERS
2------------------------------2ND LARGEST CHARACTERS
3------------------------------3RD      "         "
4------------------------------4TH      "(SMALLEST)CHARACTERS
5------------------------------SOLID VECTORS
6------------------------------DOTTED VECTORS
7------------------------------DOT-DASHED VECTORS
8------------------------------SHORT-DASHED VECTORS
9------------------------------LONG-DASHED VECTORS
```

```
                                1         NAME    CRTPLT,I$BLD1
                                2         EXT     $SE
       000001 A                 3 X       EQU     1
       000002 A                 4 B       EQU     2
       000423 A                 5 FOUR    EQU     0423
000000 000000 A                 6 CRTPLT  ENTR
                                7         REW     FCB,8
000001 006505 A
000002 000000 E
000003 100000 A
000004 001410 A
000005 000133 R
000006 000000 A
000007 000000 A
                                8         EXT     I$PLT
000010 006017 A                 9         LDAE    I$PLT
000011 000000 E
000012 001016 A                10         JANZ    AGAIN
000013 000023 R
                               11         READ    DCB1,2,,1
000014 006505 A
000015 000002 E
000016 100000 A
000017 010002 A
000020 000145 R
000021 000000 A
000022 000000 A
                               12 AGAIN   READ    FCB,8
000023 006505 A
000024 000015 E
000025 100000 A
000026 000010 A
000027 000133 R
000030 000000 A
000031 000000 A
000032 014101 A                13         LDA     FCB+1
000033 054005 A                14         STA     CALSEQ
000034 006020 A                15         LDBI    30
000035 000036 A
000036 005322 A                16 DBR     DBR
                               17         EXT     CONVRT
000037 002000 A                18         CALL    CONVRT,0
000040 000000 E
000041 000000 A
000041                         19 CALSEQ  BES     0
000042 006017 A                20         LDAE*   CALSEQ
000043 100041 R
000044 006140 A                21         SUBI    32700
000045 077674 A
000046 001002 A                22         JAP     CLSOT
000047 000061 R
000050 006017 A                23         LDAE    CALSEQ
000051 000041 R
```

```
000052 120423 A    24          ADD      FOUR
000053 006057 A    25          STAE     CALSEQ
000054 000041 R
000055 001026 A    26          JBNZ     DBR
000056 000036 R
000057 001000 A    27          JMP      AGAIN
000060 000023 R
000061 014050 A    28 CLSOT    LDA      DCBFN
000062 002016 A    29          JANZM    BUFOUT
000063 000121 R
000064 001000 A    30          RETU*    CRTPLT
000065 100000 R
000066 000000 A    31 I$BLD1   ENTR
000067 002000 A    32          CALL     $SE,1
000070 000000 E
000071 000001 A
000072 000000 A    33 ICAR     DATA     0
000073 054016 A    34          STA      LDA+1
000074 064017 A    35          STB      LDB+1
000075 074020 A    36 STX      STX      LDX+1
000076 006017 A    37          LDAE*    ICAR
000077 100072 R
000100 034031 A    38          LDX      DCBFN
000101 044030 A    39          INR      DCBFN
000102 006255 A    40          STAE     OBUF,X,0200
000103 000340 R
000104 005144 A    41          IXR
000105 005041 A    42          TXA
000106 144024 A    43          SUB      FCB
000107 002010 A    44          JAZM     BUFOUT
000110 000121 R
000111 006010 A    45 LDA      LDAI
000112 000000 A
000113 006020 A    46 LDB      LDBI
000114 000000 A
000115 006030 A    47 LDX      LDXI
000116 000000 A
000117 001000 A    48          RETU*    I$BLD1
000120 100066 R
                   49          EXT      OUTK2
000121 000000 A    50 BUFOUT   ENTR
000122 002000 A    51          CALL     OUTK2,OBUF,DCBFN
000123 000000 E
000124 000340 R
000125 000132 R
000126 005001 A    52          TZA
000127 054002 A    53          STA      DCBFN
000130 001000 A    54          RETU*    BUFOUT
000131 100121 R
                   55 *
000132 000000 A    56 DCBFN    DATA     0
                   57 FCB      FCB      120,IBUF,1
000133 000170 A
```

```
000134 000150 R
000135 000400 A
000136 000000 A
000137 000000 A
000140 000000 A
000141 000000 A
000142 000000 A
000143 000000 A
000144 000000 A
                    58 DCB1    DCB      30,IBUF
000145 000036 A
000146 000150 R
000147 000000 A
000150              59 IBUF    BSS      120
000340              60 OBUF    BSS      120
                    61         END
ENTRY NAMES
000000 R CRTPLT 000060 R I$BLD1
EXTERNAL NAMES
000070 E $SE    000040 E CONVRT 000011 E I$PLT  000123 E OUTK2
000024 E V$IOC
SYMBOLS
000070 E $SE    000023 R AGAIN  000002 A B      000121 R BUFOUT
000041 R CALSEQ 000061 R CLSOT  000040 E CONVRT 000000 R CRTPLT
000036 R DBP    000145 R DCB1   000132 R DCBFN  000133 R FCB
000423 A FOUR   000066 R I$BLD1 000011 E I$PLT  000150 R IBUF
000072 R ICAP   000111 R LDA    000113 R LDB    000115 R LDX
000340 R OBUF   000123 E OUTK2  000075 R STX    000024 E V$IOC
000001 A X
 · 0 ERRORS ASSEMBLY COMPLETE
```

```
      SUBROUTINE CONVRT(INBLK)
      DIMENSION INBLK(2,2)
      DATA IGS/0/
      DATA MAX,LX,LY,IHIY,IHIX/29700,4*-1/
      IF(INBLK(1,1).LT.0) GO TO 99
      IF(INBLK(1,1).GT.32700)GO TO 98
      IL=1
      IU=2
      IF(INBLK(1,2).EQ.32764) IU=1
      IOUT=0
      DO 3 I=IL,IU
      NX=MAX-INBLK(1,I)
      IF(NX.LT.0)NX=0
      IF(NX.GT.1430)NX=1430
      NX=I$RAST(NX)
      IF(LX.NE.NX)IOUT=1
      LX=NX
      IHIX=32+ISHIFT(NX,5,3)
      LOX=64+IAND(NX,31)
      NY=INBLK(2,I)
      IF(NY.GT.1089)NY=1089
      NY=I$RAST(NY)
      IF(LY.NE.NY)IOUT=1
      LY=NY
      IHIY=32+ISHIFT(NY,5,3)
      LOY=96+IAND(NY,31)
      IF(I.EQ.2)GO TO 56
      IF(IOUT.EQ.1)IGS=0
      IF(IGS.EQ.0)CALL I$BLD1(29)
      IF(IGS+IOUT.EQ.0)GO TO 5
      IF(IOUT.EQ.0)GO TO 55
 56   IF(IHIY.NE.KHIY)CALL I$BLD1(IHIY)
      IF(LOY.NE.KLOY.OR.IHIX.NE.KHIX)CALL I$BLD1(LOY)
      IF(KHIX.NE.IHIX)CALL I$BLD1(IHIX)
 5    CALL I$BLD1(LOX)
 55   KHIX=IHIX
      KLOY=LOY
      KHIY=IHIY
 3    IOUT=1
      IGS=1
      IF(INBLK(1,2).NE.32764)RETURN
      CALL I$BLD1(31)
      CALL I$BLD1(INBLK(2,2))
      IGS=0
      RETURN
 99   CALL I$BLD1(27)
      CALL I$BLD1(INBLK(2,2))
      RETURN
 98   IGS=0
      LX=-1
      LY=-1
      KHIY=-1
      KHIX=-1

      RETURN
      END
 0 ERRORS COMPILATION COMPLETE
/PFILE,PI,,CRTPLT
/DASMR,B
```

```
                         1          NAME     I$RAST
                         2          EXT      $SE
000000 000000 A          3 I$RAST ENTR
000001 002000 A          4          CALL     $SE
000002 000000 E
000003 000001 A          5          DATA     1
000004 000000 A          6 NX       DATA     0
000005 064014 A          7          STB      STB
000006 006027 A          8          LDBE*    NX
000007 100004 R
000010 006010 A          9          LDAI     715
000011 001313 A
000012 006160 A         10          MULI     1023
000013 001777 A
000014 006170 A         11          DIVI     1430
000015 002626 A
000016 005021 A         12          TBA
000017 024002 A         13          LDB      STB
000020 001000 A         14          RETU*    I$RAST
000021 100000 R
000022 005000 A         15 STB      NOP
                        16          END
ENTRY NAMES
000000 R I$RAST
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE      000000 R I$RAST 000004 R NX       000022 R STB
  0 ERRORS ASSEMBLY COMPLETE
```

```
                        1          NAME     TEKFNC
                        2          EXT      $SE,V$DPVE,V$DPIV
       000002 A         3 B        EQU      2
000000 000000 A         4 TEKFNC   ENTR
000001 002000 A         5          CALL     $SE,1
000002 000000 E
000003 000001 A
000004 000000 A         6 IFUNC    DATA     0
000005 054027 A         7          STA      STA
000006 064030 A         8          STB      STB
000007 006027 A         9          LDBE*    IFUNC
000010 100004 R
000011 005322 A        10          DBR
000012 005021 A        11          TBA
000013 001004 A        12          JAN      RET
000014 000034 R
000015 144035 A        13          SUB      MAX
000016 001002 A        14          JAP      RET
000017 000034 R
000020 006216 A        15          LDAE     TABLE,B,0200
000021 000042 R
000022 006020 A        16          LDBI     V$DPVE
000023 000000 E
000024 056003 A        17          STA      3,B
000025 006010 A        18          LDAI     077774
000026 077774 A
000027 056002 A        19          STA      2,B
000030 005301 A        20          DECR     01
000031 056000 A        21          STA      0,B
000032 002000 A        22          CALL     V$DPIV
000033 000000 E
000034 006010 A        23 RET      LDAI     0
000035 000000 A
000035                 24 STA      BES      0
000036 006020 A        25          LDBI     0
000037 000000 A
000037                 26 STB      BES      0
000040 001000 A        27          RETU*    TEKFNC
000041 100000 R
000042 000070 A        28 TABLE    DATA     56,57,58,59,96,97,98,99,100
000043 000071 A
000044 000072 A
000045 000073 A
000046 000140 A
000047 000141 A
000050 000142 A
000051 000143 A
000052 000144 A
000053 000011 A        29 MAX      DATA     9
                       30          END
ENTRY NAMES
000000 R TEKFNC
EXTERNAL NAMES

000002 E $SE     000033 E V$DPIV 000023 E V$DPVE
SYMBOLS
000002 E $SE     000002 A B        000004 R IFUNC   000053 R MAX
000034 R RET     000035 R STA      000037 R STB     000042 R TABLE
000000 R TEKFNC  000033 E V$DPIV   000023 E V$DPVE
  0 ERRORS ASSEMBLY COMPLETE
```

5. MAGNETOGRAM PRODUCTION FROM MAGNETOMETER NETWORK

A program to produce magnetograms from the AFGL Magnetometer Network data was written. The program is to be run on the Varian computer and the magnetograms are hard copies of a Cathode Ray Tube (CRT) display.

The program is run by typing in on the CRT keyboard

/LOAD, DEMO

The program DEMO then prompts the programmer

ENTER START HOUR OR 99 FOR BOT.

93 FOR CURRENT TAPE POSITION, 97 FOR DISK DATA

If you want to begin the magnetogram at a particular hour enter that hour as hh. If the magnetogram is to begin at the beginning of the TAPE (BOT) enter 99, if it is to begin from the current position of the tape enter 93. If the magnetogram is to be made from data already on disk enter 97.

ENTER START TIME FOR GRAPH IF DIFFERENT

The magnetograms should begin on the hour, but if the data to be plotted do not begin on the hour, enter a starting hour. For example say the data begin at 17:37, the programmer may enter 17 as the starting hour of the magnetogram. otherwise enter RETURN

ENTER START DAY IF DIFFERENT FROM CURRENT TAPE DAY

If the tape is positioned on day 213 for example and you want a magnetogram beginning on day 214 then enter 214. Otherwise enter RETURN.

ENTER NUMBER OF HOURS AND NUMBER OF PERIODS IF NOT 1 (HHPP)

Enter the length of the magnetograms in hours and the desired number of magnetograms. If you want seven eight hour magnetograms enter 0708. If only eight hour magnetogram is wanted enter 08.

AUTO HARD COPY (Y,N)?

If you want hard copies of the magnetograms produced automatically enter Y, if not enter N.

INDIVIDUAL PLOTS, OVERLAY PLOT, OR BOTH (I,O,B)?

If you want an individual plot of each station and no overlay

enter I, if you want an overlay and no individual plots enter O. If you want both enter B.

DEMO is a highly automated program. Once all the entries have been made all that is required of the programmer is that he change the tapes if the magnetogram requires more than one tape and that he terminate the program. The program is terminated by simply entering a / RETURN.

An example of the prompting sequence and the resulting magnetograms is shown in Figures 3 thru 9. These are the plots of X, Y, and Z components of the seven network stations, in order: MA, FL, MI, WI, SD, CA, WA. Figure 10 is a composite of all seven stations.

Figures 1 and 2 are plots of LINCHK for the selected date of the magnetograms, and give an overall view (for that date) of the quality of operation of the network for each station.

<u>To use LINCHK</u>

Load the tape of interest and type on the CRT keyboard /LOAD,LINCHK

Each plot contains seven miniature plots , 1 for each station. The abscissa is the time of day in hours, for the day (or days) listed in the heading. In the top plot the ordinate represents the percent of frames for each station which contains errors; in the bottom plot the ordinate represents the percent of frames for each station which is missing. On each miniature plot, the base line refers to zero percent of the frames. The ordinate can take any value from 0-100%, where the value of 100% for a given station would lie just below the baseline of the station just above it.

```
ENTER START HOUR OR 99 FOR BOT,
 93 FOR CURRENT TAPE POSITION, 97 FOR DISK DATA
08
 ENTER START TIME FOR GRAPH IF DIFFERENT

 ENTER START DAY IF DIFFERENT FROM CURRENT TAPE DAY
155
 ENTER NUMBER OF HOURS AND # OF PERIODS IF NOT 1 (HHPP)
12
 AUTO HARD COPY (Y,N)?
Y
 INDIVIDUAL PLOTS, OVERLAY PLOT, OR BOTH (I,O,B)?
B
```

8048/00:04 TO 8048/12:46 FILE 2

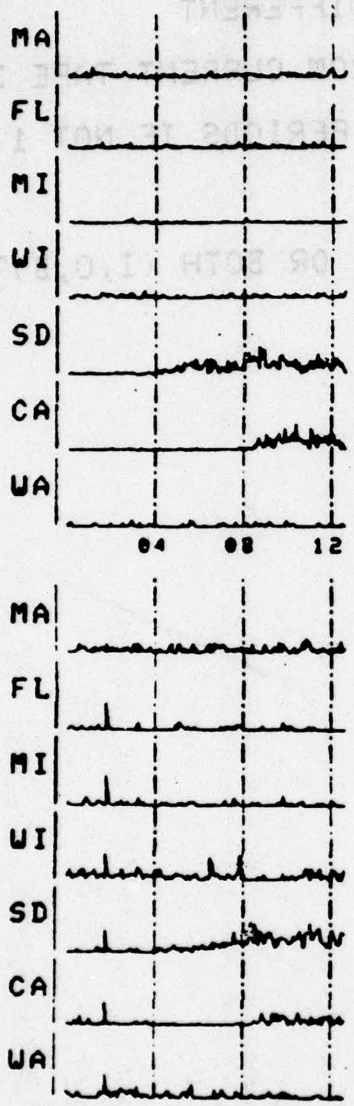


Figure 1.

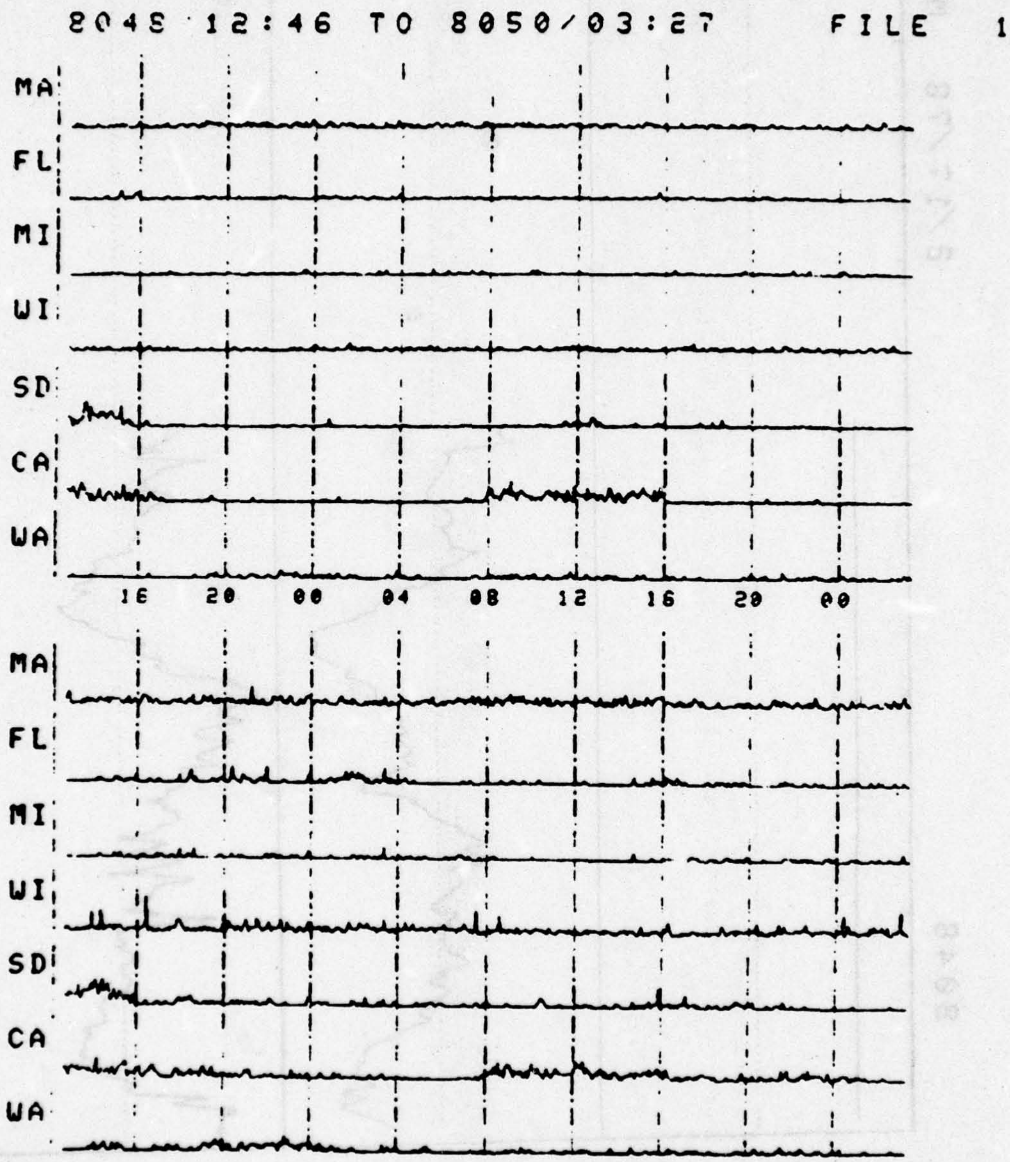
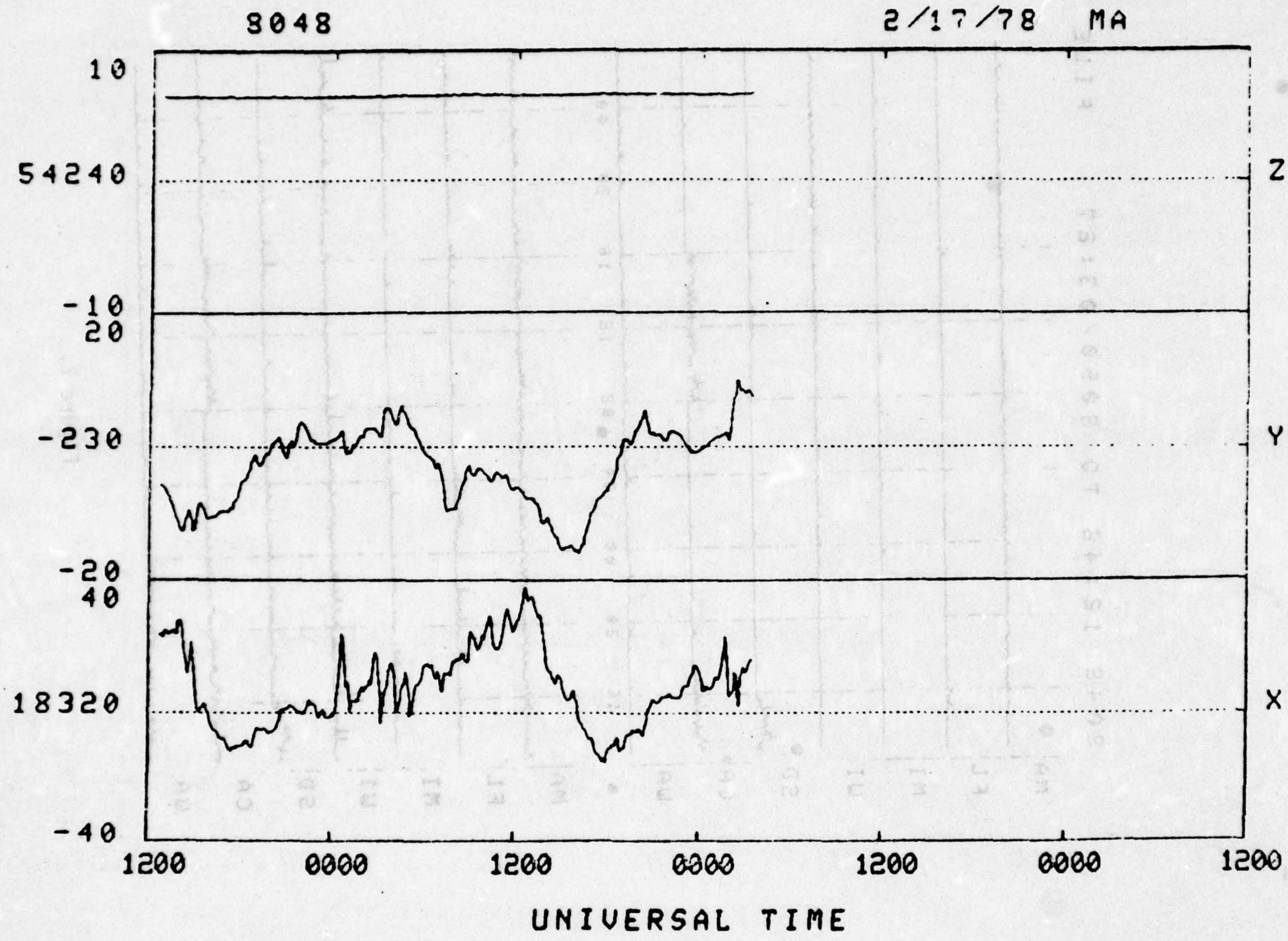

8048 12:46 TO 8050/03:27 FILE 1
MA
FL
MI
WI
SD
CA
WA
16 20 00 04 08 12 16 20 00
MA
FL
MI
WI
SD
CA
WA

Figure 2.



8048
2/17/78 MA
10
54240
-10
20
-230
-20
40
18320
-40
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 3.

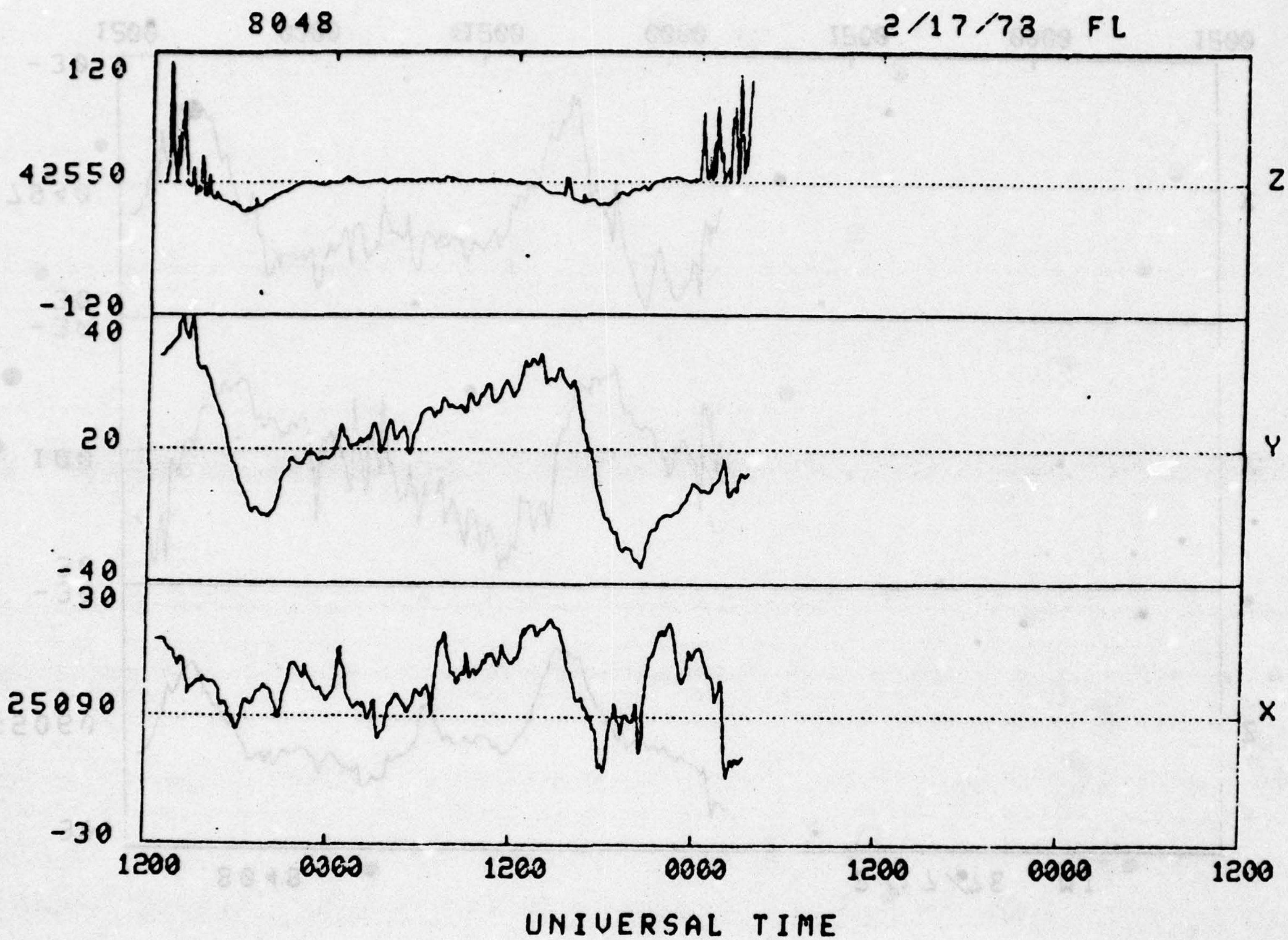
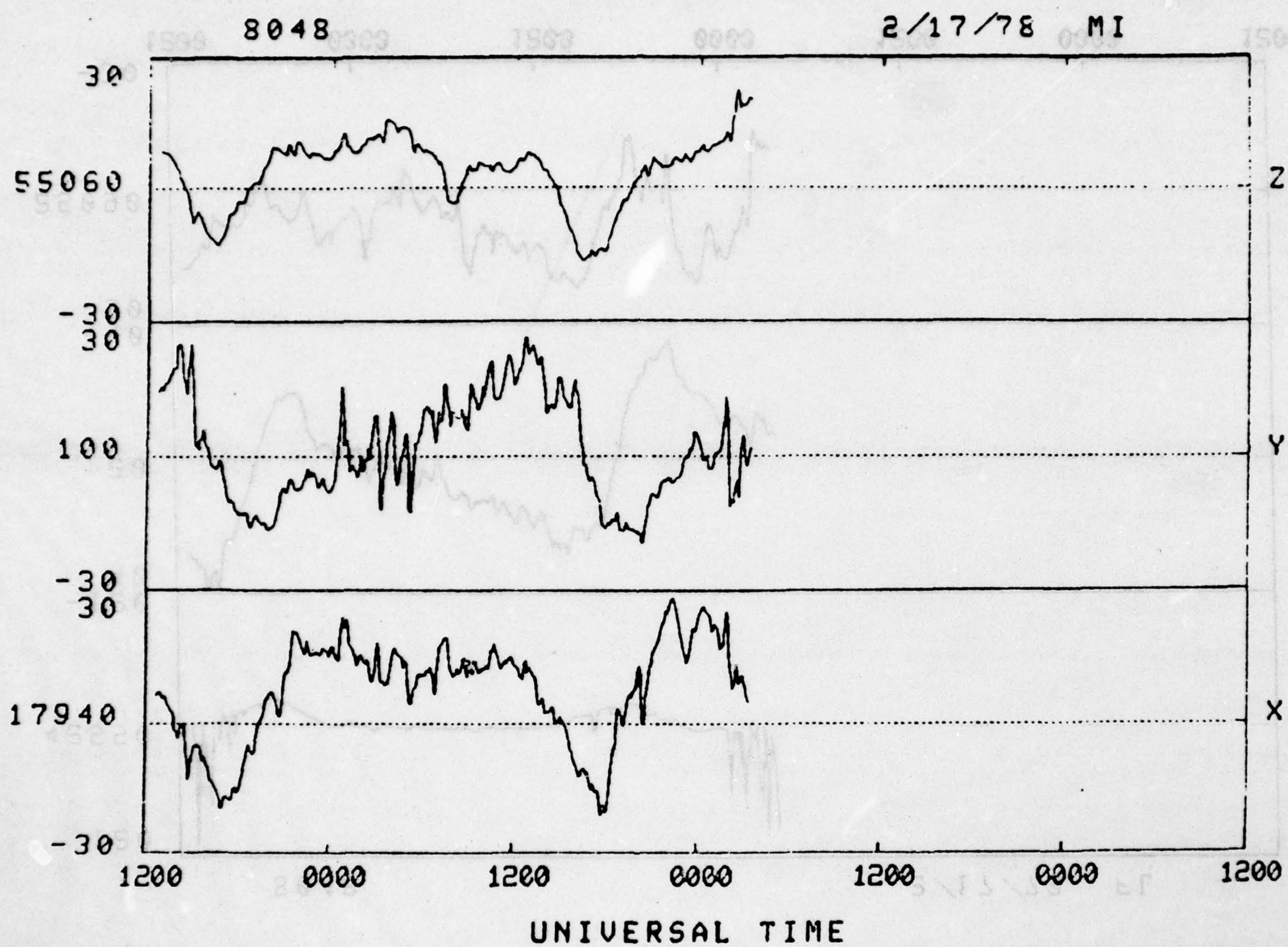

8048
2/17/78 FL
120
42550
-120
40
20
-40
30
25090
-30
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 4.



8048
2/17/78 MI
30
55060
-30
30
100
-30
30
17940
-30
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 5.

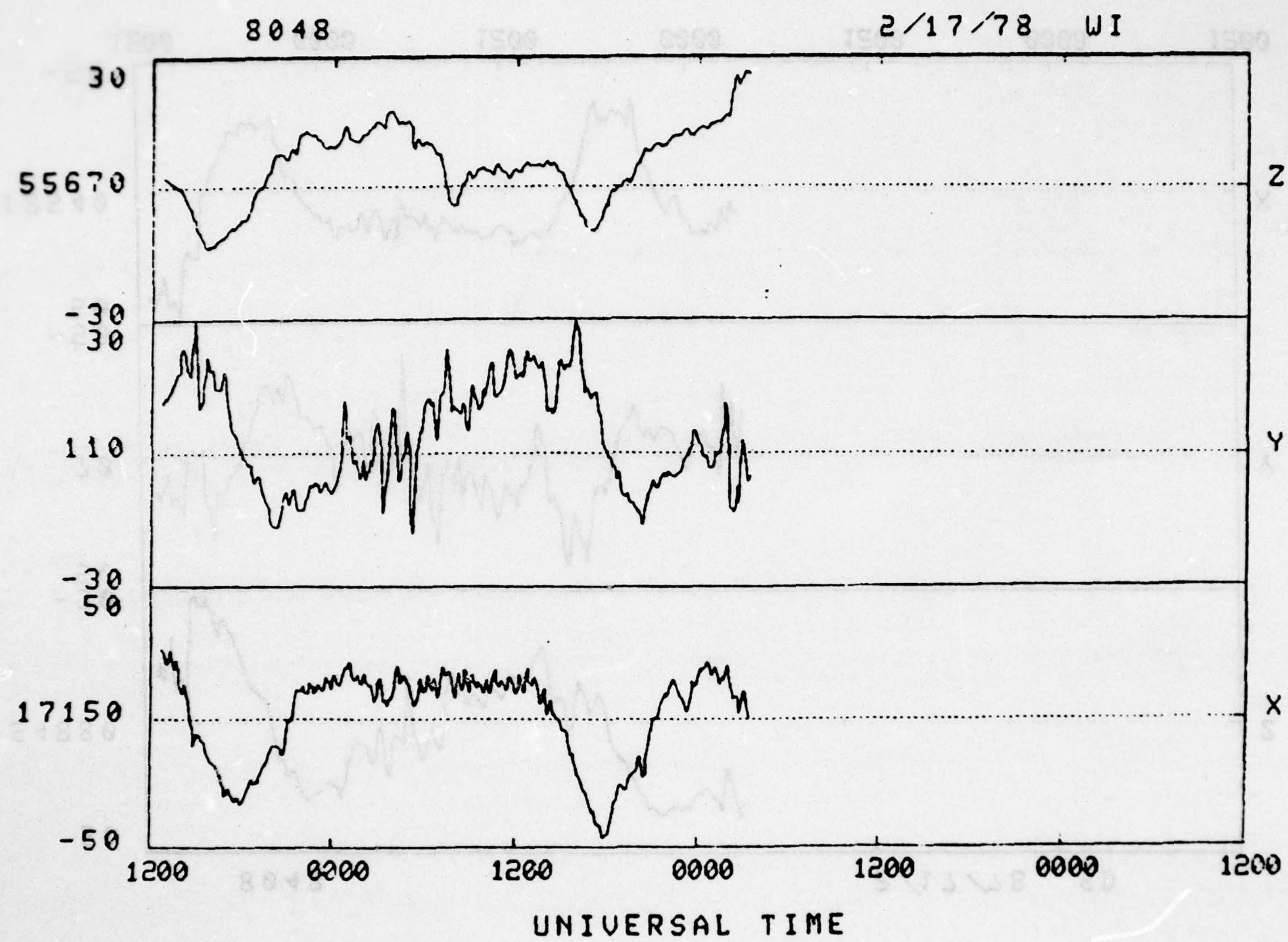

8048
2/17/78 WI
30
55670
-30
30
110
-30
50
17150
-50
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 6.

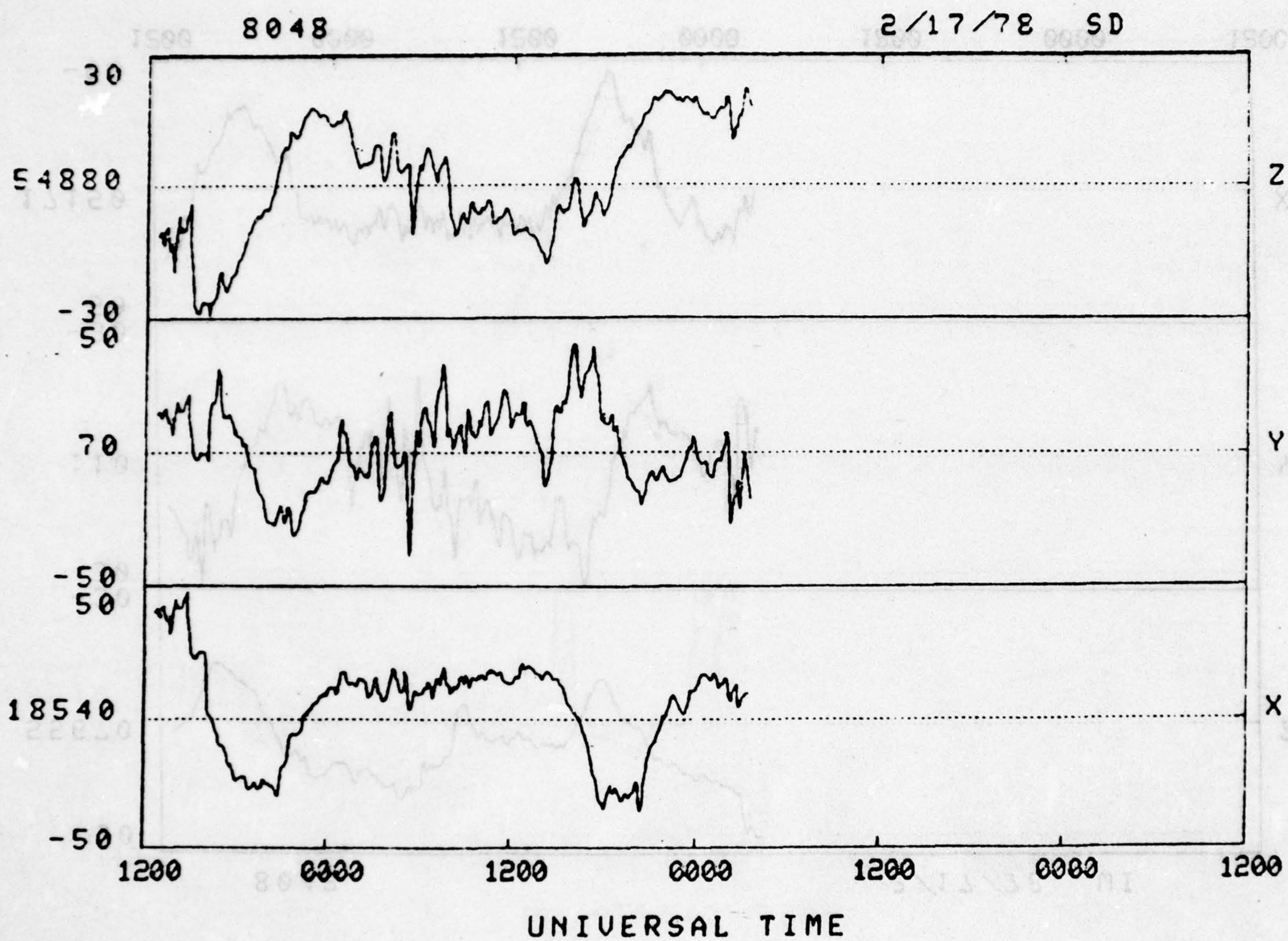

8048
2/17/78 SD
30
54880
-30
50
70
-50
50
18540
-50
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 7.

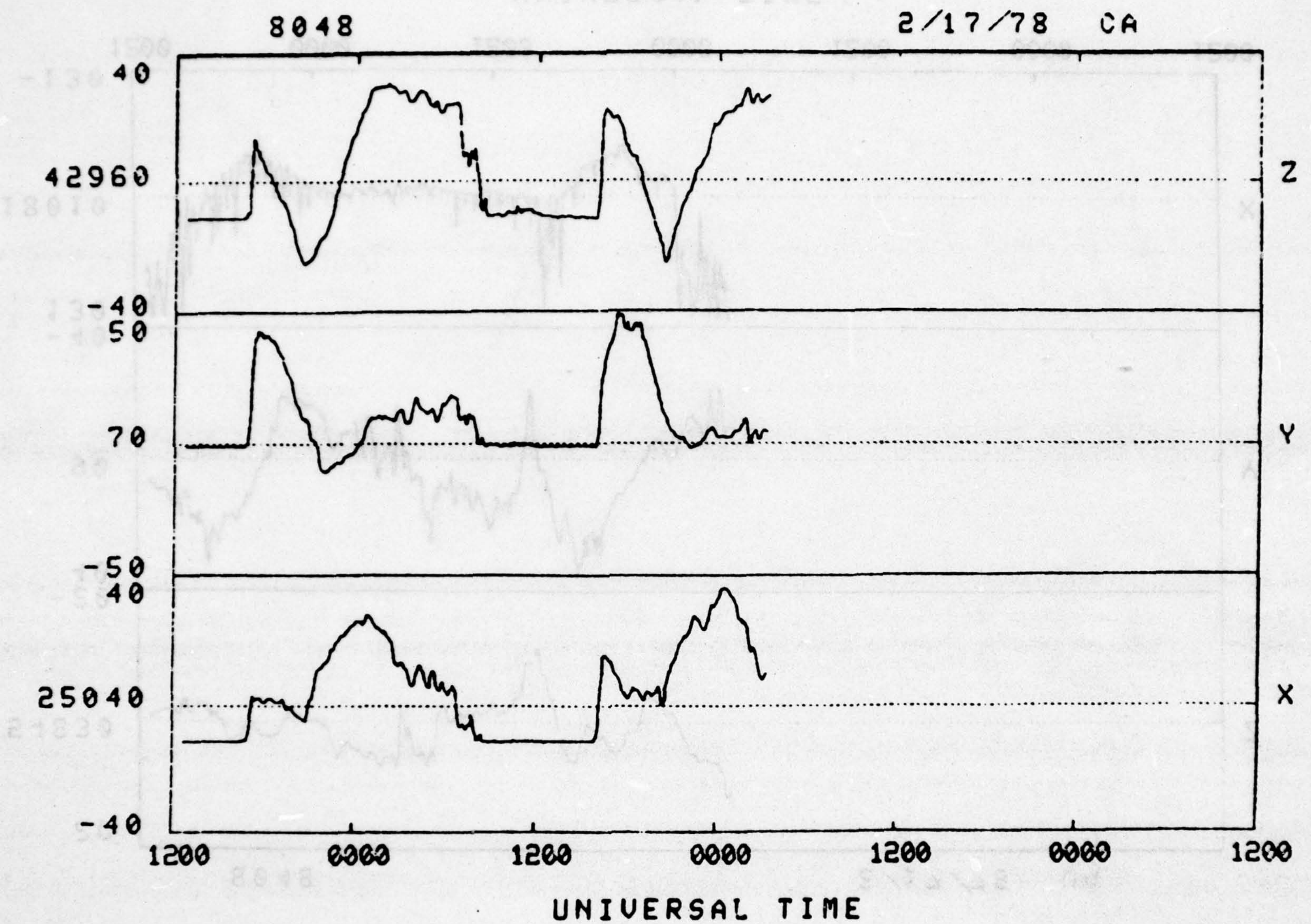

8048
2/17/78 CA
40
42960
-40
50
70
-50
40
25040
-40
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 8.

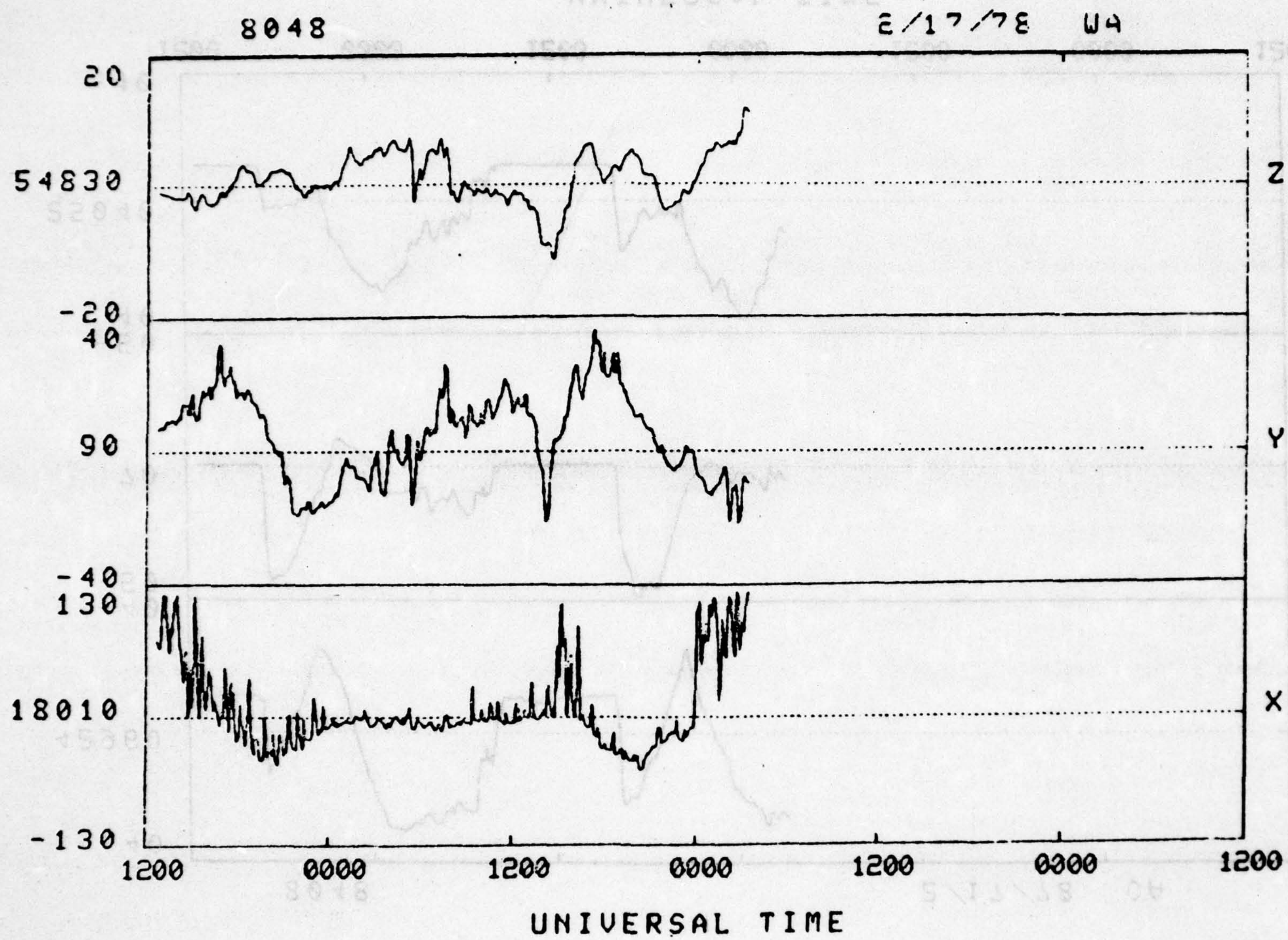

8048
2/17/78 WA
20
54830
-20
40
90
-40
130
18010
-130
Z
Y
X
1200
0000
1200
0000
1200
0000
1200
UNIVERSAL TIME

Figure 9.

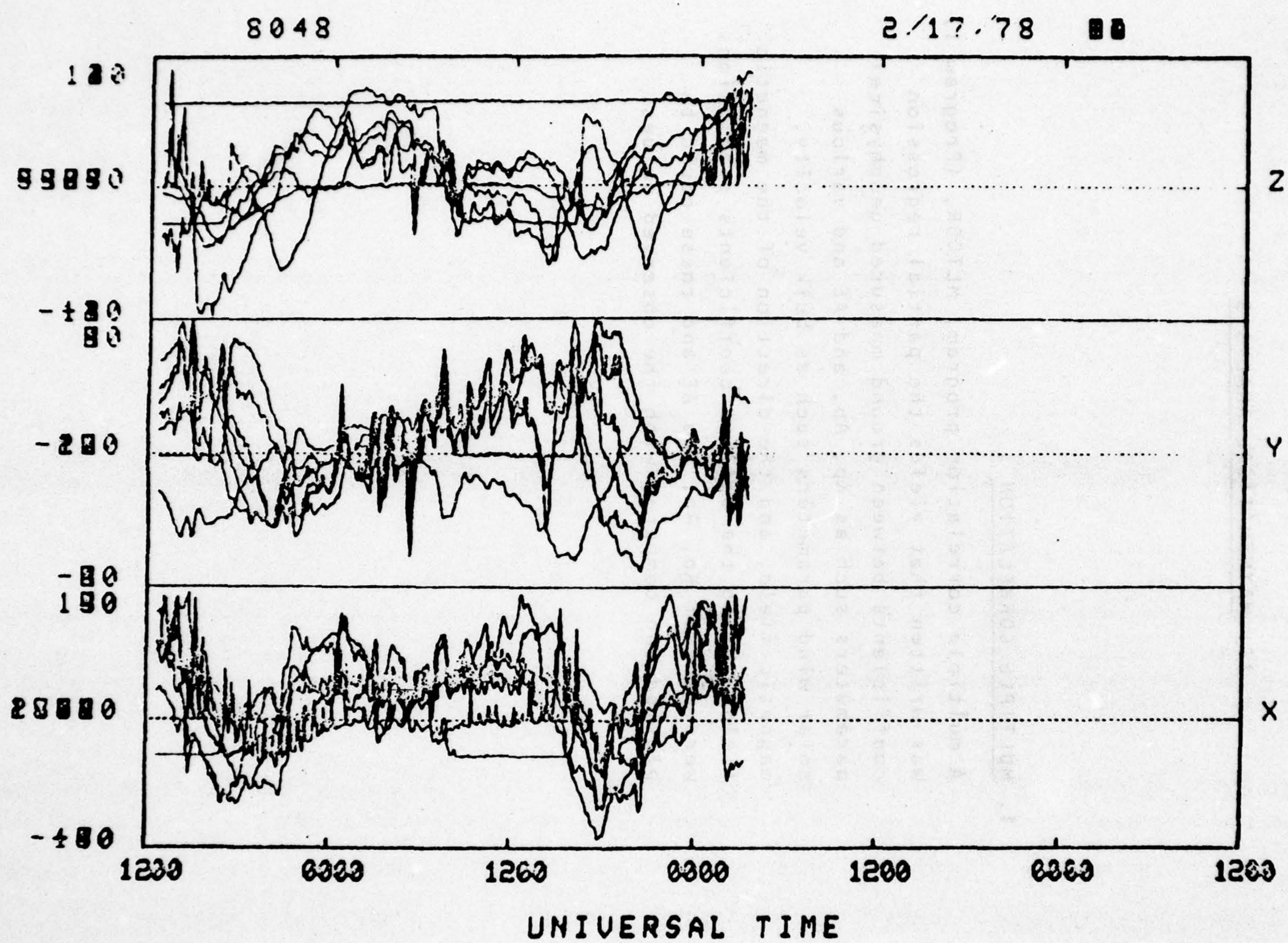


Figure 10.

II. MATHEMATICAL ANALYSIS

1. MULTIPLE CORRELATION

A multiple correlation program, MLTCOR, (Program 1) was written that yields the partial regression coefficients between ground measured geophysical parameters such as Kp, Ap, and AE and various solar wind parameters such as bulk velocity, magnetic field, and the direction of the magnetic field. Using the derived coefficients predictions were made of Kp, Ap, and AE and these could be plotted and compared with the observed values.

Program 1

```
BELMS,CM110000,T150.                                        2261    BELLEW
ATTACH,TAPE1,CORDTX3218,ID=BELLEW,MR=1.
ATTACH,TAPE2,IMFX3218,ID=BELLEW,MR=1.
ATTACH,PEN,ONLINEPEN,MR=1.
FTN,SL,PL=77777.
LDSET(LIB=PEN)
LGO.
```

```
      PROGRAM MLTCOR(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE6=OUTPU
     +T)

C        CATALOGED AS MLTCORX3218,CY=5

      DIMENSION B(6),SB(6)
      DIMENSION PROGID(3)
      COMMON/SUB/B,SB,SY,BNOT
      COMMON/DAT/LAG,JMPDT
      DATA PROGID/6RBELLEW,4R2261,2R89/
      CALL PLTID3(PROGID,200.0,11.0,0.75)
      F=2.5
      TOL=.0001
      N=6
      M=100
      SIGN=-1
      JMPDT=0
      DO 2 J=1,2
      LAG=1
      DO 1 I=1,1
      CALL SETUP(N,M,SIGN)
      CALL SUB1(F,TOL,M)
      LAG=LAG+1
      JMPDT=JMPDT+1
      CALL ENDPLT
      STOP
      END
```

```
      SUBROUTINE SETUP(N,M,SIGN)

      SETUP READS DATA FROM TAPE1 AND CONSTRUCTS
      A MATRIX OF THE CROSS CORRELATION COEFFICIENTS

      DIMENSION A(6,6),Y(600,6),YB(6),SIG(6),R(6,6)
      COMMON/SET/YB,SIG,R,NM,DF,Y
      COMMON/DAT/LAG,JMPDT
      COMMON/MUL/KNT
      REWIND 3
      PRINT 100
      FORMAT(1H1)
      FORMAT()
      PRINT 300, SIGN,LAG
      FORMAT(/*       SIGN=*,F5.1,*         LAG=*,I2///)
          N NUMBER OF DEPENDENT AND INDEPENDENT VARIABLES
          M NUMBER OF OBSERVATIONS
      MSAV=M
      CALL DATIN(M)
      PI=2.0*ACOS(0.0)
      FP=1.0/(4*PI)
      REWIND 3
      NM=N-1
      DF=M-1
      KNT=0
      DO 1 J=1,M
      READ(3) IDATE,B,BX,BY,BZ,SGBX,SGBY,SGBZ,
     +VTH,RHO,V,VX,VY,VZ,AE
      PRINT 75,B,BX,BY,BZ,VTH,RHO,V,VX,VY,VZ,AE
      FORMAT(4F10.4/7F10.4)
      IF(SIGN*BZ.LT.0.0) GO TO 1
      KNT=KNT+1
      Y(KNT,1)=RHO*V*V
      SX=FP*((BY*BY+BZ*BZ)*VX-(VY*BY+VZ*BZ)*BX)
      SY=FP*((BX*BX+BZ*BZ)*VY-(VX*BX+VZ*BZ)*BY)
      SZ=FP*((BX*BX+BY*BY)*VZ-(VX*BX+VY*BY)*BZ)
      S=SQRT(SX*SX+SY*SY+SZ*SZ)
      PRINT 11,S,SX,SY,SZ
      Y(KNT,2)=S
      Y(KNT,2)=BZ
      Y(KNT,3)=V*BZ
      Y(KNT,5)=SQRT(SGBX*SGBX+SGBY*SGBY+SGBZ*SGBZ)
      Y(KNT,4)=BZ*BZ*V
      Y(KNT,2)=Y(KNT,3)*Y(KNT,5)
      Y(KNT,6)=AE
      CONTINUE
      M=KNT
      PRINT 15,N,M
      FORMAT(/2I10/)
      PRINT 11,((Y(J,I),I=1,N),J=1,M)
      PRINT 200
```

```
11    FORMAT(1X,1P6E12.3)
C        DO 6 FINDS THE MEAN OF EACH VARIABLE
      DO 6 I=1,N
      YB(I)=0
      DO 5 J=1,M
5     YB(I)=YB(I)+Y(J,I)
6     YB(I)=YB(I)/M
      PRINT 11,(YB(I),I=1,N)
      PRINT 200
C        DO 7 FINDS THE DEVIATIONS
      DO 7 I=1,N
      DO 7 J=1,M
7     Y(J,I)=Y(J,I)-YB(I)
      PRINT 12,((Y(J,I),I=1,N),J=1,M)
      PRINT 200
C        DO 4 CONSTRUCTS A MATRIX OF SQUARES AND
C        CROSS PRODUCTS OF DEVIATIONS
      DO 4 I=1,N
      DO 4 K=I,N
      A(I,K)=0
      DO 8 J=1,M
8     A(I,K)=A(I,K)+Y(J,I)*Y(J,K)
      A(K,I)=A(I,K)
4     CONTINUE
C     PRINT 12,((A(J,I),I=1,N),J=1,N)
      PRINT 200
12    FORMAT(1P6E12.3).
C
      DO 13 I=1,N
13    SIG(I)=SQRT(A(I,I))
      PRINT 11,(SIG(I),I=1,N)
      PRINT 200
      DO 14 I=1,N
      DO 14 J=I,N
      R(I,J)=A(I,J)/(SIG(I)*SIG(J))
14    R(J,I)=R(I,J)
      PRINT 12,((R(I,J),J=1,N),I=1,N)
      M=MSAV
C
      RETURN
      END
```

```
SUBROUTINE SUB1(F,TOL,M)

DIMENSION B(6),SB(6)
DIMENSION YB(6),SIG(6),R(6,6)
DIMENSION Y(600,6)
COMMON/SUB/B,SB,SY,BNOT
COMMON/SET/YB,SIG,R,NM,DF,Y

PRINT 200
FORMAT()
CLRL=0
F1=F
N=NM+1
F2=F
PRINT 940,F1,F2
FORMAT(*     F1,F2 =*1P2E12.3)
VMIN=10**10
VMAX=0.0
NMAX=0
NMIN=0
SY=SIG(N)*SQRT(R(N,N)/DF)
PRINT 120,SY
FORMAT(*      SY=*F10.5)
DO 10 J=1,NM
B(J)=0

I=1
CONTINUE
   PRINT*,R(I,I),I,I

   THIS TEST REDUCES THE POSSIBILITY OF DEGENERACY
   WHEN AN INDEPENDENT VARIABLE IS APPROXIMATELY A
   LINEAR COMBINATION OF OTHER INDEPENDENT VARIABLES.
IF(R(I,I).LE.TOL) GO TO 3

   LINEAR COMBINATION OF OTHER INDEPENDENT VARIABLES
VI=R(I,N)*R(N,I)/R(I,I)
   PRINT*,VI,VMAX,I
IF(VI.EQ.0) GO TO 3
IF(VI.LT.0.0) GO TO 1
IF(VI.LE.VMAX) GO TO 3
VMAX=VI
NMAX=I
   PRINT*,VMAX,NMAX
GO TO 3
B(I)=SIG(N)/SIG(I)*R(I,N)
   PRINT*,B(I),I
SB(I)=SY/SIG(I)*SQRT(1.0/R(I,I))
SB(I)=SY/SIG(I)*SQRT(R(I,I))
PRINT 430,B(I),SB(I),I
FORMAT(1P2E12.3,0PI10)
```

```
      IF(ABS(VI).GE.ABS(VMIN)) GO TO 3
      VMIN=VI
      NMIN=I
      IF(I.EQ.NM) GO TO 100
      I=I+1
      GO TO 4
      SUM=0
      DO 101 J=1,I
      SUM=SUM+B(J)*YB(J)
      BNOT=YB(N)-SUM
      IF(CLRL.NE.0) CALL COMLC(B)
      FIN=ABS(VMIN)*DF/R(N,N)
      PRINT 710,FIN,F2,BNOT
      FORMAT(*      FIN,F2,BNOT =*1P3E12.3)
      IF(FIN.GE.F2) GO TO 6
      K=NMIN
      PRINT 620,K
      FORMAT(*      K=NMIN=*I2)
      DF=DF+1
      GO TO 5
      CONTINUE
      FOUT=VMAX*(DF-1.0)/(R(N,N)-VMAX)
      PRINT 711,FOUT,F1
      FORMAT(*      FOUT,F1 =*1P2E12.3)
      CONTINUE
      IF(VMAX*(DF-1.0)/(R(N,N)-VMAX).LE.F1) RETURN
      K=NMAX
      PRINT 123,K
      FORMAT(*      K=NMAX=*I2)
      DF=DF-1
      CALL NUMAT(K,N)
      CLRL=1
      GO TO 7

      END
```

```
SUBROUTINE NUMAT(K,N)
DIMENSION YB(6),SIG(6),R(6,6),D(6,6)
DIMENSION Y(600,6)
COMMON/SET/YB,SIG,R,NM,DF,Y
PRINT 200
PRINT 50,((R(I,J),J=1,N),I=1,N)
FORMAT(1P6E12.3)
FORMAT()
DO 1 I=1,N
DO 1 J=1,N
IF(I.EQ.K.AND.J.EQ.K) GO TO 10
IF(I.EQ.K.AND.J.NE.K) GO TO 9
IF(I.NE.K.AND.J.EQ.K) GO TO 8
D(I,J)=(R(I,J)*R(K,K)-R(I,K)*R(K,J))/R(K,K)
GO TO 1
D(I,J)=-R(I,K)/R(K,K)
GO TO 1
D(I,J)=R(K,J)/R(K,K)
GO TO 1
D(I,J)=1.0/R(I,J)
CONTINUE
DO 2 I=1,N
DO 2 J=1,N
R(I,J)=D(I,J)
PRINT 200
PRINT 50,((R(I,J),J=1,N),I=1,N)
RETURN
END
```

```
SUBROUTINE DATIN(M)
DIMENSION X(14),IX(14),ID(14)
COMMON/DAT/LAG,JMPDT
IF(JMPDT.NE.0) GO TO 10
REWIND 1
REWIND 2
REWIND 3
LAGL=LAG
LAGM=1
LLAG=LAG-1
KOUNT=0
READ(1) IDATE1,(X(I),I=1,13)
IF(EOF(1)) 5,2
READ(2) IDATE2,IX11
IF(EOF(2)) 7,9
X(14)=IX11
IF(IDATE1.GT.IDATE2) GO TO 2
IF(IDATE2.EQ.IDATE1) GO TO 4
READ(1) IDATE1,(X(I),I=1,13)
IF(IDATE1.GT.IDATE2) GO TO 2
IF(EOF(1)) 5,3
CONTINUE
IF(LAG.EQ.0) GO TO 12
IF(LAGL.EQ.LAG.OR.LAG.EQ.1) GO TO 18
DO 17 I=1,LLAG
ID(I)=ID(I+1)
IX(I)=IX(I+1)
CONTINUE
I=LAG
GO TO 22
I=1
READ(2) ID(I),IX(I)
IX11=IX(I)
IDD1=IDATE1-(IDATE1/100)*100
IDD2=ID(I)-(ID(I)/100)*100
IDIF=IDD2-IDD1
IF(IDIF.EQ.LAG.OR.IDIF.EQ.LAG-24) GO TO 19
IF(I.EQ.LAG.AND.LAGL.EQ.LAG) GO TO 19
IF(ID(I)-IDATE1.EQ.LAG) GO TO 19
I=I+1
GO TO 22
LAGM=LAG
X(14)=IX11
WRITE(3) IDATE1,(X(I),I=1,14)
KOUNT=KOUNT+1
PRINT 54, IDATE1,(X(I),I=1,7)
FORMAT(I10,7F10.3)
PRINT 56,(X(I),I=8,14)
FORMAT(7F10.3)
IF(KOUNT.EQ.M) RETURN
LAGL=1
```

```
      IF(LAG.EQ.0) GO TO 1
      I=1
      READ(1) IDATE1,(X(J),J=1,13)
      IF(EOF(1)) 5,55
      CONTINUE
      IF(IDATE1.EQ.ID(I)) GO TO 4
      I=I+1
      IF(I.LE.LAG) GO TO 55
      LAGM=1
      LAGL=LAG
      GO TO 2
      PRINT 6
      FORMAT(*      EOF(1)*)
      RETURN
      PRINT 8
      FORMAT(*      EOF(2)*)
      RETURN
      END
      SUBROUTINE COMLC(B)

         THIS SUBROUTINE CALCULATES THE TOTAL CORRELATION
         COEFFICIENT OF THE OBSERVED DEPENDENT VARIABLE Y,
         AND THE PRWDICTED DEPENDENT VARIABLE YJ.

      DIMENSION YB(6),SIG(6),R(6,6),Y(600,6)
      DIMENSION B(6)
      DIMENSION PY(100),PX(100)
      COMMON/SET/YB,SIG,R,NM,DF,Y
      COMMON/MUL/M
      N=NM+1
      CYJY=0
      SIGYJ=0
      DO 2 I=1,M
      YJ=0
      DO 1 J=1,NM
      YJ=YJ+B(J)*Y(I,J)
      PY(I)=Y(I,N)
      PX(I)=YJ
      CYJY=CYJY+YJ*Y(I,N)
      SIGYJ=SIGYJ+YJ*YJ
      SIGYJ=SQRT(SIGYJ)
      RML=CYJY/(SIGYJ*SIG(N))
      P=0
      DO 3 I=1,NM
      IF(B(I).NE.0) P=P+1
      CONTINUE
      RR=RML*RML
      FTEST=RR/(1-RR)*(M-P-1)/P
         PRINT*,"   RMUL,FTEST,M,P=",RML,FTEST,M,P
      CALL SPLOT(PX,PY,M,99,XMAX,YMAX,X0,Y0)
      CALL PLOTTER(PX,PY,-1000.0,1000.0,5.0,0.0,6,11,1,M)
      RETURN
      END
```

AD-A066 344 EMMANUEL COLL BOSTON MASS F/G 9/2
INVESTIGATION OF MICROPULSATION ACTIVITY.(U)
NOV 78 W F BELLEW, M P HAGAN, R L VESPRINI F19628-76-C-0013
UNCLASSIFIED AFGL-TR-78-0312 NL

2 OF 2
AD
A066344

END
DATE
FILMED
5--79
DDC

```
      SUBROUTINE SPLOT(X,Y,N,ISCAL,XMAX,YMAX,X0,Y0)
      DIMENSION X(1),Y(1),G(101,61)
      DATA BLANK/1H /,AXIS/1H./,POINT/1H*/
***
* X(I)= ARRAY FOR X-AXIS
* Y(I)= ARRAY FOR Y-AXIS
* N= NUMBER OF POINTS
* X0,Y0= ORIGIN OF GRAPH
* DX,DY= SCALE FACTOR MULTIPLYING POINT VALUES
* **NOTE**TOO HIGH OR TOO LOW X,Y WILL CAUSE ERROR IF ISCAL.NE.
     +99
* IF ISCAL=99 THEN DO AUTO-SCALING
      IF(ISCAL.NE.99) GO TO 99
***
      X0=XMAX=X(1)
      Y0=YMAX=Y(1)
      DO 98 K=2,N
      IF(X(K).LT.X0) X0=X(K)
      IF(Y(K).LT.Y0) Y0=Y(K)
      IF(X(K).GT.XMAX) XMAX=X(K)
      IF(Y(K).GT.YMAX) YMAX=Y(K)
 98   CONTINUE
 99   CONTINUE
      DX=100./(XMAX-X0)
      DY=60./(YMAX-Y0)
      DO 1 I=1,101
 1    G(I,1)=AXIS
      DO 7 I=1,61
 7    G(1,I)=AXIS
      DO 2 I=2,101
      DO 2 J=2,61
 2    G(I,J)=BLANK
* PLOT POINTS
      DO 3 I=1,N
* 1. IS ADDED SO X=0 DOES NOT GIVE A 0 INDEX
      IX=(X(I)-X0)*DX+1.
      IY=(Y(I)-Y0)*DY+1.
 3    G(IX,IY)=POINT
      DELX=1./DX
      DELY=1./DY
      PRINT 6, X0,XMAX,DELX,Y0,YMAX,DELY,N
 6    FORMAT(1H1,10X,*XMIN,XMAX,DELX=*,3F10.3,2X,
     +*YMIN,YMAX,DELY=*,3F10.3,2X,*NO. PTS.=*,I6)
      DO 4 I=1,61
* STEP NECESSARY SO Y-AXIS COMES OUT RIGHT SIDE UP
      II=62-I
 4    PRINT 100,(G(J,II),J=1,101)
 100  FORMAT (1H ,10X,101A1)
      RETURN
      END
```

```
      SUBROUTINE PLOTTER(X,BF,XB,XE,XO,YO,NDX,NNX,NORM,N)
      DIMENSION X(1),BF(1)
      COMMON/YMB/YM,B
         IF(NORM.EQ.0) GO TO 3
      CALL SLIN(BF,NORM,FB,FS,N)
         GO TO 2
         YM=1.0
         B=5.0
      CALL BOXER(XO,YO,10.0,10.0,XB,XE,NDX,NNX)
         PRINT 10
         FORMAT(*      BOXER CALLED*)
      EX=(X(1)-XB)*10.0/(XE-XB)
         IF(EX.LT.0.0.OR.EX.GT.10.0) GO TO 25
      Y=YM*BF(1)+B
      CALL PLOT(EX,Y,3)
      CALL SYMBOL(EX,Y,.2,1,0.0,-1)
         PRINT 9,X(1),BF(1),EX,Y
      GO TO 21
      CALL PLOT(0,0,3)
      DO 1 I=2,N
      EX=(X(I)-XB)*10.0/(XE-XB)
         IF(EX.LT.0.0.OR.EX.GT.10.0) GO TO 1
      Y=YM*BF(I)+B
         IF(MOD(I,50).EQ.0) PRINT 9,X(I),BF(I),EX,Y,I
         FORMAT(2(0PF12.4,1PE13.4),I10)
      CALL PLOT(EX,Y,2)
      CALL SYMBOL(EX,Y,.2,1,0.0,-1)
      CONTINUE
         RETURN
      END
```

```
      SUBROUTINE BOXER(XO,YO,XL,YL,XB,XE,NDX,NYX)
      COMMON/YMB/YM,B
      COMMON /CONST/PI,XC,ALPHA,XMINUS,X1,X2,X3,X4,XPLUS,XON,XN
     +U,XM
     1,XN,DEL,DEL2,A,EXPA,XNU2,ALFA4,ALFA2,ALFA42
      CALL PLOT(XO,YO,-3)
      CALL PLOT(XL,0,2)
      CALL PLOT(XL,YL,2)
      CALL PLOT(0,YL,2)
      CALL PLOT(0,0,2)
      SIGN=1.0
      YB=0.0
      DY=YB+SIGN*.12
      X=XL/NDX
      LU=NDX-1
         WRITE(6,130) X,LU
         FORMAT(F12.5,I10)
      DO 1 I=1,LU
      CALL PLOT(X,YB,3)
      CALL PLOT(X,DY,2)
      X=X+XL/NDX
      IF(SIGN.LT.0.0) GO TO 5
      SIGN=-SIGN
      YB=10.0
      GO TO 4
      XFI=0.0
         WRITE(6,110)
         FORMAT(*          X-AXIS COMPLETED    *)
      SIGN=1.0
      DX=XFI+SIGN*.12
      Y=.5
      DO 2 I=1,19
      CALL PLOT(XFI,Y,3)
      CALL PLOT(DX,Y,2)
      Y=Y+.5
      IF(SIGN.LT.0.0) GO TO 7
      SIGN=-SIGN
      XFI=XL
      GO TO 6
      CONTINUE
         WRITE(6,120)
         FORMAT(*          Y-AXIS COMPLETED    *)
      CALL PLOT(XO,YO,3)
      Y=-1.0
      X=1.0
      DO 3 I=1,9
      XI=X*(XE-XB)/XL+XB
      IF(XI.LT.XO) GO TO 3
      SQ=SQRT(XO/XI)
      DEG=57.2958*ACOS(SQ)
         PRINT 10,X,XI,DEG,I
```

```
   FORMAT(3F12.5,I10)
CALL NUMBER(X,Y,.15,DEG,0.0,1)
X=X+1.0
XNUM=XB
Y=-.5
X=0
DO 8 I=1,NNX
XNUM=X*(XE-XB)/XL+XB
PRINT 102,X,XNUM
FORMAT(2F12.5)
CALL NUMBER(X,Y,.15,XNUM,0.0,1)
X=X+XL/(NNX-1)
X=-1.0
Y=0
YNUM=(Y-B)/YM
DO 9 I=1,11
CALL NUMBER(X,Y,.15,YNUM,0.0,1)
   PRINT 10,X,Y,YNUM,I
Y=Y+1.0
YNUM=(Y-B)/YM
RETURN
END
```

```
SUBROUTINE SLIN(F,NORM,FB,FS,N)
DIMENSION F(1)
COMMON/YMB/YM,B
FB=F(1)
FS=F(1)
DO 1 I=2,N
IF(F(I).GT.FB) FB=F(I)
IF(F(I).LT.FS) FS=F(I)
CONTINUE
IF(NORM.NE.1) GO TO 25
IF(ABS(FS).GT.FB) FB=FS
DO 20 I=1,N
F(I)=F(I)/FB
B=0.0
YM=10.0
   PRINT 12,FB,FS,YM,B
IF(FS.LT.0.0) GO TO 10
RETURN
YM=5.0
B=5.0
   PRINT 12,FB,FS,YM,B
RETURN
IF(FS.LT.0) GO TO 2
B=0
U=.00001
IF(U.GT.FB) GO TO 3
U=U*10.0
GO TO 4
YM=10.0/U
IF(YM*FB.LE.5.0) YM=2.0*YM
   PRINT 12,FB,FS,YM,B
RETURN
IF(FB.LT.0) GO TO 7
B=5.0
IF(ABS(FS).GT.FB) FB=ABS(FS)
U=.00001
IF(U.GT.FB) GO TO 5
U=U*10.0
GO TO 6
YM=5.0/U
IF(YM*FB.LE.2.5) YM=2.0*YM
   PRINT 12,FB,FS,YM,B
RETURN
B=10.0
FB=ABS(FS)
U=.00001
IF(U.GT.FB) GO TO 8
U=U*10.0
GO TO 9
YM=10.0/U
IF(YM*FB.LE.5) YM=2.0*YM
   PRINT 12,FB,FS,YM,B
   FORMAT(4F12.5)
RETURN
END
```

2. COMPARISON OF SOLAR-WIND PARAMETERS AND GEOMAGNETIC ACTIVITY INDICES

Philip M. Pazich

William F. Bellew

March 1976

Magnetospheric Dynamics Branch

Space Physics Division

I. DATA

A comparison of solar-wind parameters and geomagnetic indices was made to determine the parameters that could best be used for prediction purposes. The solar-wind data consisted of 1000 hourly averages of plasma (v and ρ) and magnetic field data (Bx, By, Bz, σx, σy, σz) from the Explorer 33 satellite. The data were measured between 19 January and 6 April 1968. An earlier study by Garrett et al. (1974) also used data from the Explorer 33 satellite measured during this period. Approximately half of the data used in the current study coincided with the data set of Garrett et al.

Solar-wind data for each hour were recorded on magnetic tape along with the AE index for the same hour and the following two hours so that correlations could be done easily. The Kp and ap indices were read from an additional data file as needed. A given hour of solar-wind data was correlated with the Kp and ap indices for both the three-hour interval in which the solar-wind data were measured and the following three-hour interval.

II. CORRELATIONS

Functions of solar-wind variables were determined on the basis of physical arguments and previous studies to be those that gave the highest correlations with magnetic activity indices.

The indices are:

AE 0, AE 1, AE 2	-	Auroral Electrojet index lagged 0, 1, and 2 hours
Kp 0, Kp 3 ap 0, ap 3	-	Kp and ap indices from the same 3-hour interval as the solar-wind data (Kp 0, ap 0) and from the following 3-hour interval (kp 3, ap 3)

The solar-wind variables are:

BZSE	-	magnetic field component Bz, SE coordinates (gammas)
BZSM	-	" " " " SM " "
RHO	-	solar wind ion density ($\#/cm^3$)
v	-	solar wind velocity (km/sec)
SIGT or ST	-	$(\sigma_x^2 + \sigma_y^2 + \sigma_z^2)^{\frac{1}{2}}$ for the hourly average

Correlations between the magnetic indices and functions determined from the solar-wind variables are listed in Table I. Brackets indicate that only negative values of the quantity were used; values > 0 were set equal to 0.

III. MULTIPLE REGRESSION

A multiple linear regression was performed using the functions and indices listed in Table I. The highest, consistent correlations were for the functions $v \cdot \langle BZSM \rangle$ and $v \cdot SIGT$ for indices AE 1, Kp 0, ap 0. The regression equations for these three indices as determined from the entire data set are given in Table II. The function $\langle BZSM \rangle$ is the solar magnetospheric Bz component when that component is negative and is set equal to zero for positive values of the component. In the 1000 hours of data

there were 509 hours with an average negative Bz and 491 hours with an average positive Bz. Additional terms to the regression equation did not significantly improve the multiple correlation coefficient.

The data was divided into two 500 hour segments to determine the effect of a specific data sample on the regression. The results of these regression analyses are listed in Table III. For the Kp and ap equations, the solar-wind functions were selected in opposite order by the regression procedure for the two data segments. This indicates that the inclusion of the additional term serves only to increase the accuracy of the fit for the specific data.

The regression analyses for the AE index lagged by one hour always gave the highest multiple correlation coefficient and the computed coefficients in the regression equation were the most consistent between data samples. An equation for Kp and ap determined by multiple linear regression on specific data of v, ⟨BZSM⟩, and SIGT could be expected to result in correlation coefficients of between .5 and .6 when applied to another data set. An equation to predict AE 1 would give a somewhat better correlation of between .6 and .7 when applied to a different data set

REFERENCE

Garrett, H.B., A.J. Dessler, and T.W. Hill, Influence of solar wind variability on geomagnetic activity, JGR, 79, No. 31, November 1974, p. 4603.

Table I. CORRELATIONS FOR 1000 HOURLY AVERAGES OF SOLAR-WIND DATA AND GEOMAGNETIC INDICES

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	AE 1	-																			
2	ap 0	.521	-																		
3	Kp 0	.638	.878	-																	
4	-⟨BZSM⟩	.660	.403	.466	-																
5	-v·⟨BZSM⟩	.700	.418	.492	.982	-															
6	v·⟨BZSM⟩^{2}	.542	.404	.419	.919	.888	-														
7	-⟨BZSM⟩·ST	.499	.451	.447	.813	.795	.841	-													
8	-v·⟨BZSM⟩·ST	.555	.470	.479	.816	.828	.823	.984	-												
9	SIGT	.217	.468	.440	.070	.085	.124	.372	.387	-											
10	v·SIGT	.228	.463	.457	.028	.016	.029	.235	.278	.939	-										
11	-v·BZSM	.627	.262	.362	.816	.836	.673	.637	.671	.044	.117	-									
12	-v·BZSE	.563	.254	.311	.654	.688	.534	.488	.532	.012	.021	.846	-								
13	-BZSM	.620	.279	.376	.850	.848	.718	.669	.684	.035	.106	.984	.330	-							
14	-BZSE	.550	.248	.307	.678	.695	.566	.504	.534	.016	.020	.830	.985	.842	-						
15	RHO·v^2	.227	.402	.390	.258	.210	.303	.350	.316	.373	.254	.030	.054	.054	.062	-					
16	-⟨BZSE⟩	.592	.313	.383	.782	.792	.730	.594	.621	.037	.001	.682	.793	.694	.809	.156	-				
17	AE 0	.765	.599	.699	.557	.594	.466	.428	.478	.237	.249	.519	.461	.515	.447	.270	.483	-			
18	AE 2	.770	.407	.534	.531	.557	.437	.402	.441	.173	.181	.513	.439	.511	.434	.172	.457	.533	-		
19	Kp 3	.620	.560	.678	.477	.498	.438	.435	.466	.398	.411	.361	.299	.373	.297	.369	.386	.544	.679	-	
20	ap 3	.582	.564	.602	.475	.490	.479	.483	.506	.359	.370	.351	.297	.365	.293	.387	.388	.504	.632	.917	-

Table II. MULTIPLE LINEAR REGRESSION FOR 1000 HOURS OF SOLAR-WIND DATA

R= Multiple Correlation Coefficient

Kp 0 =	$-.00219\ v\cdot\langle BZSM\rangle + 5.43$	R= 0.492
=	$-.00216\ v\cdot\langle BZSM\rangle + .00382\ v\cdot SIGT + 2.68$	R= 0.666
ap 0 =	$.0117\ v\cdot SIGT + 2.99$	R= 0.463
=	$.0115\ v\cdot SIGT - .00546\ v\ BZSM + .00319$	R= 0.618
AE 1 =	$-.146\ v\cdot\langle BZSM\rangle + 114.5$	R= 0.699
=	$-.145\ v\cdot\langle BZSM\rangle + .0863\ v\cdot SIGT + 52.3$	R= 0.732

Table III. MULTIPLE LINEAR REGRESSION FOR TWO SEGMENTS OF DATA

a) First 500 Hours

$Kp\ 0 = -.00242\ v\cdot\langle BZSM\rangle + 5.23$ R= 0.470

$ = -.00249\ v\cdot\langle BZSM\rangle + .00344\ v\cdot SIGT + 2.54$

$ap\ 0 = -.00572\ v\cdot\langle BZSM\rangle + 78.5$ R= 0.450

$ = -.00590\ v\cdot\langle BZSM\rangle + .00817\ v\cdot SIGT + 1.47$ R= 0.621

$AE\ 1 = -.140\ v\cdot\langle BZSM\rangle + 102.9$ R= 0.631

$ = -.142\ v\cdot\langle BZSM\rangle + .101\ v\cdot SIGT + 23.6$ R= 0.701

b) Second 500 Hours

$Kp\ 0 = .00510\ v\cdot SIGT + 3.59$ R= 0.532

$ = .00467\ v\cdot SIGT - .00183\ v\cdot\langle BZSM\rangle + 2.61$ R= 0.695

$ap\ 0 = .0185\ v\cdot SIGT + .0140$ R= 0.569

$ = .0175\ v\cdot SIGT - .00459\ v\cdot\langle BZSM\rangle - 2.46$ R= 0.658

$AE\ 1 = -.145\ v\cdot\langle BZSM\rangle + 129.5$ R= 0.731

$ = -.142\ v\cdot\langle BZSM\rangle + .0767\ v\cdot SIGT + 79.7$ R= 0.749

3. SPECTRAL ANALYSIS

The second area of data handling was in preparing micropulsation data for power spectral analysis. The completed programs from this effort

a. take the data from the archive tape and convert it into engineering units and give a quick plot of the data (Program 2) and

b. filter the data so that it will be in a form suitable for spectral analysis (Program 3).

Work is going on to produce a contour plotting routine to display dynamic spectra, that is, spectral density contours will be plotted on a time - frequency graph.

PROGRAM 2

This program unpacks data from the archive tape and plots it on the printer for a quick look.

The subroutine LEVELER suppresses points that differ from the average by more than one RMS. This is to ensure that the whole plot will not be dominated by a few large spurious values.

PROGRAM 3

In order to properly frequency analyze a time series it must first be filtered to avoid aliasing. We assume what we are analyzing is a continuous process in time, however it is sampled at discrete intervals, say $\Delta\tau$.

For instance let the power spectrum, $P(f)$, of a stationary process, $X(t)$, be

$$P(f) = \int_{-\infty}^{\infty} C(\tau) e^{-i\omega\tau} d\tau \qquad (\omega = 2\pi f)$$

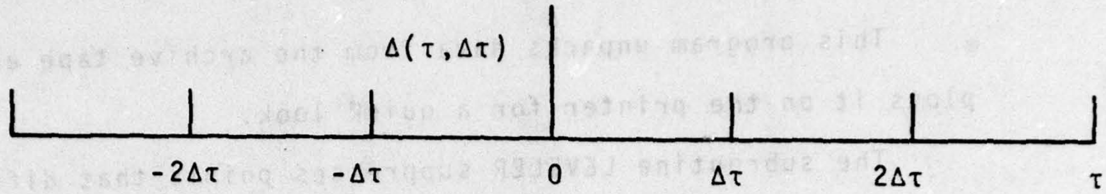
where

$$C(\tau) = \lim_{T\to\infty} \int_{-T}^{T} X(t)\, X(t+\tau) dt$$

is the auto covariance. Let the values of $C(\tau)$ be given only at uniformly spaced values of τ, $\tau=0$, $\pm\Delta\tau$, $\pm 2\Delta\tau$ $\pm$.... we can calculate the aliased power spectrum $Pa(f)$ by

$$Pa(f) = \int_{-\infty}^{\infty} \nabla(\tau,\Delta\tau)\ C(\tau) e^{-i\omega\tau} d\tau \qquad (1)$$

where $\nabla(\tau,\Delta\tau)$ is an infinite Dirac comb



The teeth of the comb are δ-functions. Applying the convolution theorem to equation (1) we get

$$Pa(f) = A(f;1/\Delta\tau)\cdot P(f)$$

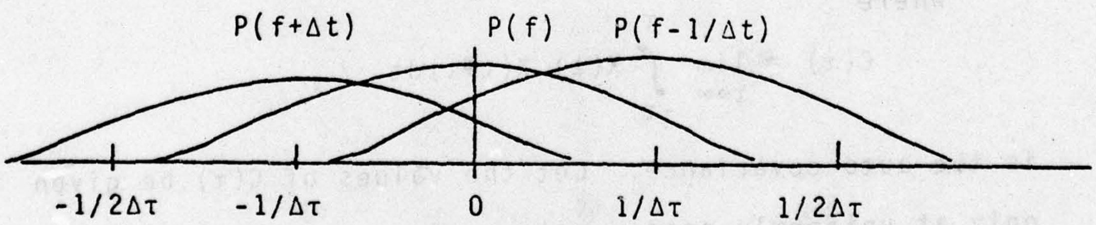
where A is also an infinite Dirac comb

$$A(f;1/\Delta t) = \sum_{q=-\infty}^{q=\infty} \delta(f-q/\Delta t)$$

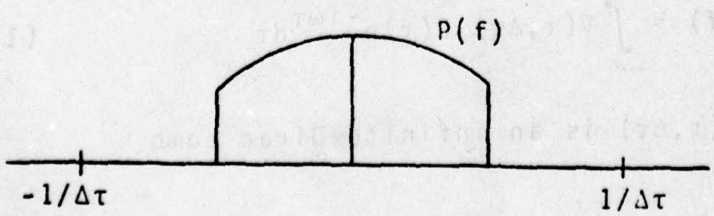
Explicitly we have

$$Pa(f) = \sum_{q=-\infty}^{q=\infty} P(f-q/\Delta t) \qquad (2)$$

From eq.(2) it is easy to see how high frequency components make contributions to low frequency components in the spectrum, see Fig. 1



The only way to eliminate aliasing is for Pa(f) to cut off for $f \geq 1/2\Delta t$



With this cut off Pa(f) is equal to P(f) for $f < f_N = 1/2\Delta\tau$, this frequency is defined as the Nyquist frequency.

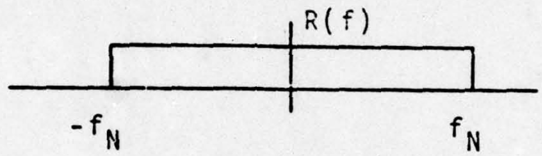
To avoid aliasing the stationary time series I(t) must be passed through a filter $F(\tau)$

$$\phi(t) = \int_{\infty}^{\infty} F(\tau) I(t-\tau) d\tau$$

Transforming to frequency space we have

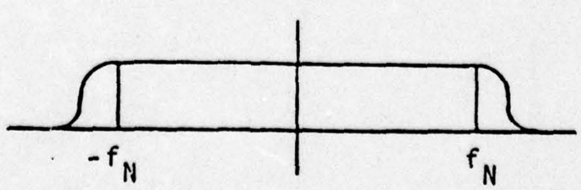
$$\phi(f) = R(f) I(f)$$

If $\phi(f)$ is to reproduce $I(f)$ exactly for $f < f_N$ then R(f), the filter response must be of the form



To achieve such a filter response requires an infinitely long filter in the time domain (This is analogous to a finite wave packet in space requiring an infinite number of frequency components). Taking only a finite amount of data - letting the filter cut off after some time leads to Gibbs phenomenon, that is the filter response oscillates in the neighborhood of the cut off frequency.

This can be mitigated by not requiring a sharp cut off. This is affected in Program 3 by replacing the sharp cut off by a sine-terminator, adding half a sine wave to the filter response

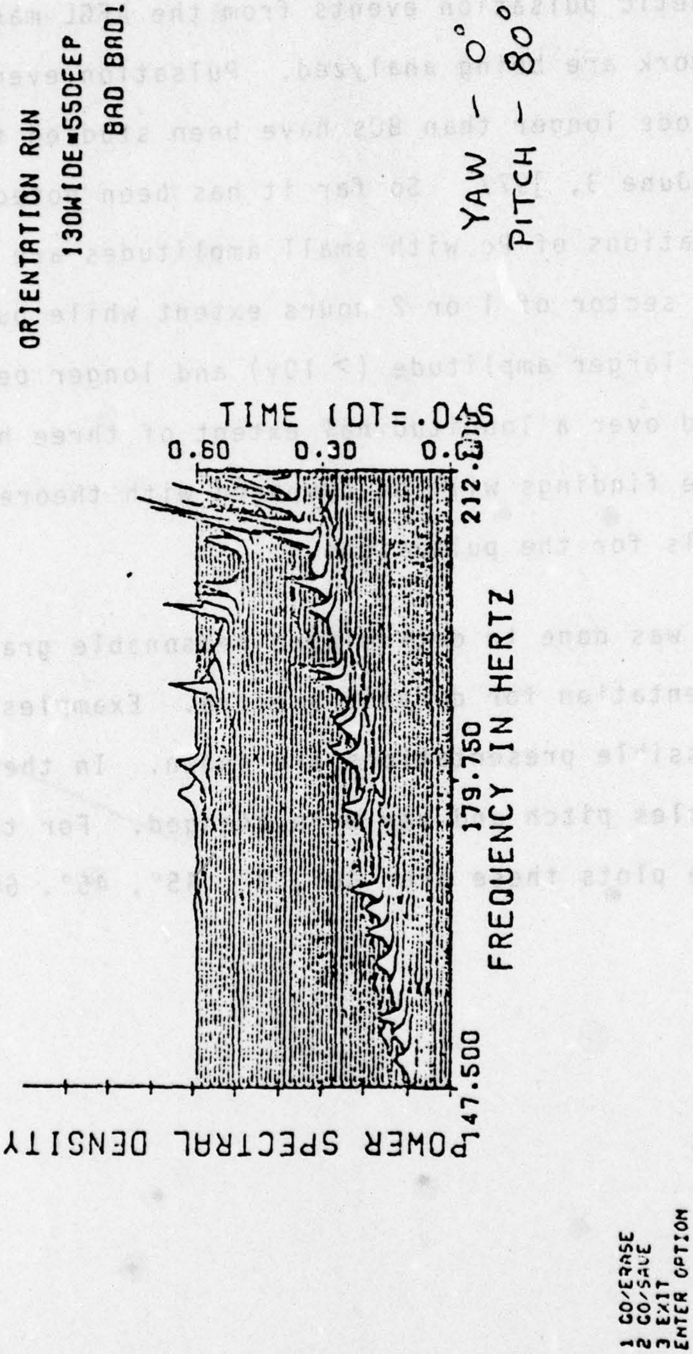


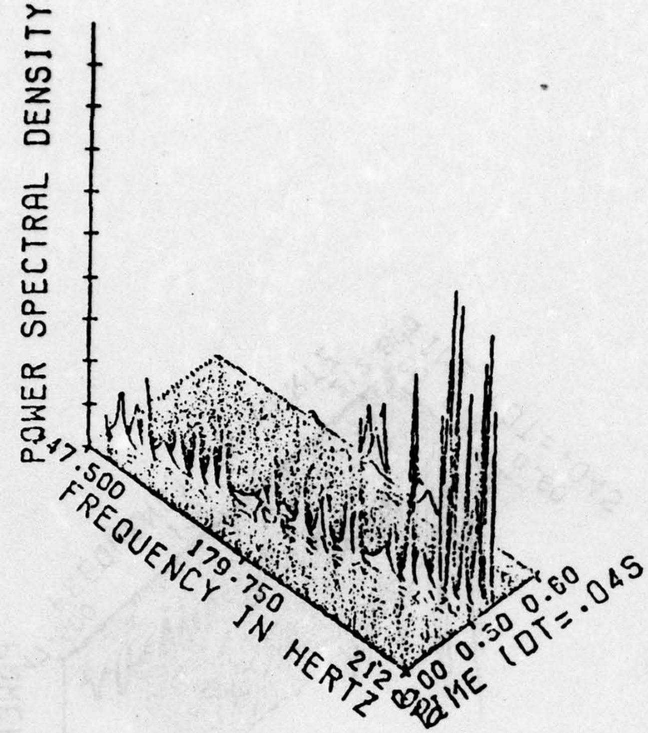
The program then calculates the weights $W_n = F(k\Delta\tau)$ that make up the filter. This filter is described in a NASA report by Behannon and Ness (Design of Numerical Filters for Geomagnetic Data Analysis, NASA TN-D-3341).

4. ANALYSIS OF MAGNETIC PULSATION EVENTS

Magnetic pulsation events from the AFGL magnetometer network are being analyzed. Pulsation events with periods longer than 80s have been studied for the day June 3, 1977. So far it has been noted that pulsations of Pc with small amplitudes are located in a sector of 1 or 2 hours extent while pulsations with larger amplitude ($> 10\gamma$) and longer period are found over a longitudinal extent of three hours. These findings will be compared with theoretical models for the pulsations.

Work was done to determine a reasonable graphics presentation for dynamic spectra. Examples of 3 possible presentations are shown. In these examples pitch and yaw were changed. For the three plots these are: 80°, 0°, 45°, 45°, 60°, 45°.

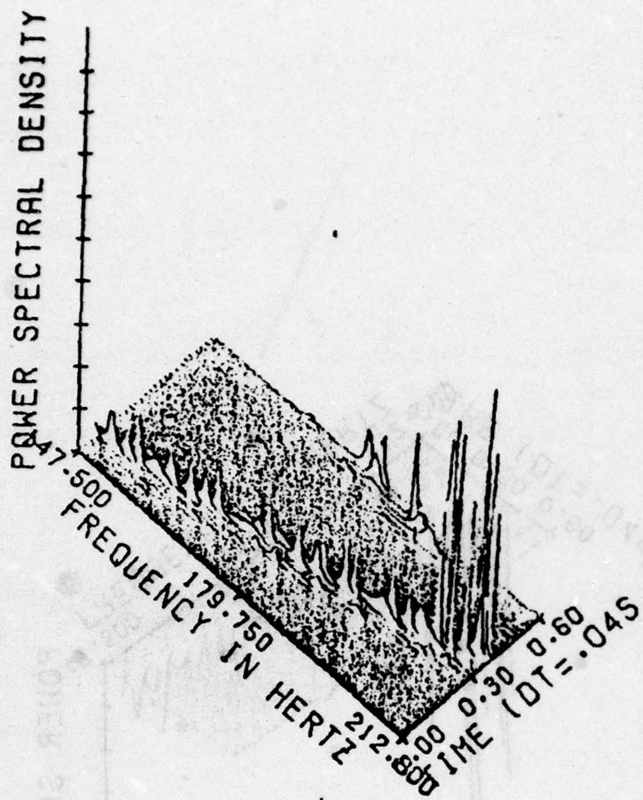

ORIENTATION RUN
130WIDE 55DEEP
BAD BAD!
YAW – 0°
PITCH – 80°
TIME (DT=.04S
0.60
0.30
0.00
212.000
179.750
147.500
FREQUENCY IN HERTZ
POWER SPECTRAL DENSITY
1 GO/ERASE
2 GO/SAVE
3 EXIT
ENTER OPTION



ORIENTATION RUN
130WIDE*55DEEP
BAD BAD!

YAW - 45°
PITCH - 45°

1 GO/ERASE
2 GO/SAVE
3 EXIT
ENTER OPTION



ORIENTATION RUN
130WIDE*550EEP
BAD BAD1

YAW – 45°
PITCH – 60°

1 GO/ERASE
2 GO/SAVE
3 EXIT
ENTER OPTION

Work was begun to speed up the program that filters the data before they are spectrally analyzed. Since the output of a filter is a convolution

$$O(x) = \int F(x')I(x-x')dx'$$

its Fourier transform is simply the product of the Fourier transform of the filter and the Fourier transform of the data. So to find the output we need only find the inverse Fourier transform of this product. It is by means of the Fast Fourier Transform that we hope to speed up the filtering.

5. A Study of Micropulsation Events

A study has been undertaken to determine how best to use data from the MAGAF Network to specify and predict status of the magnetosphere by means of micropulsations.

A a first step a number of magnetically quiet days on which all seven stations were active has been selected for study to establish the quiet time behavior of micropulsations. A list of these days and some representative magnetograms are given.

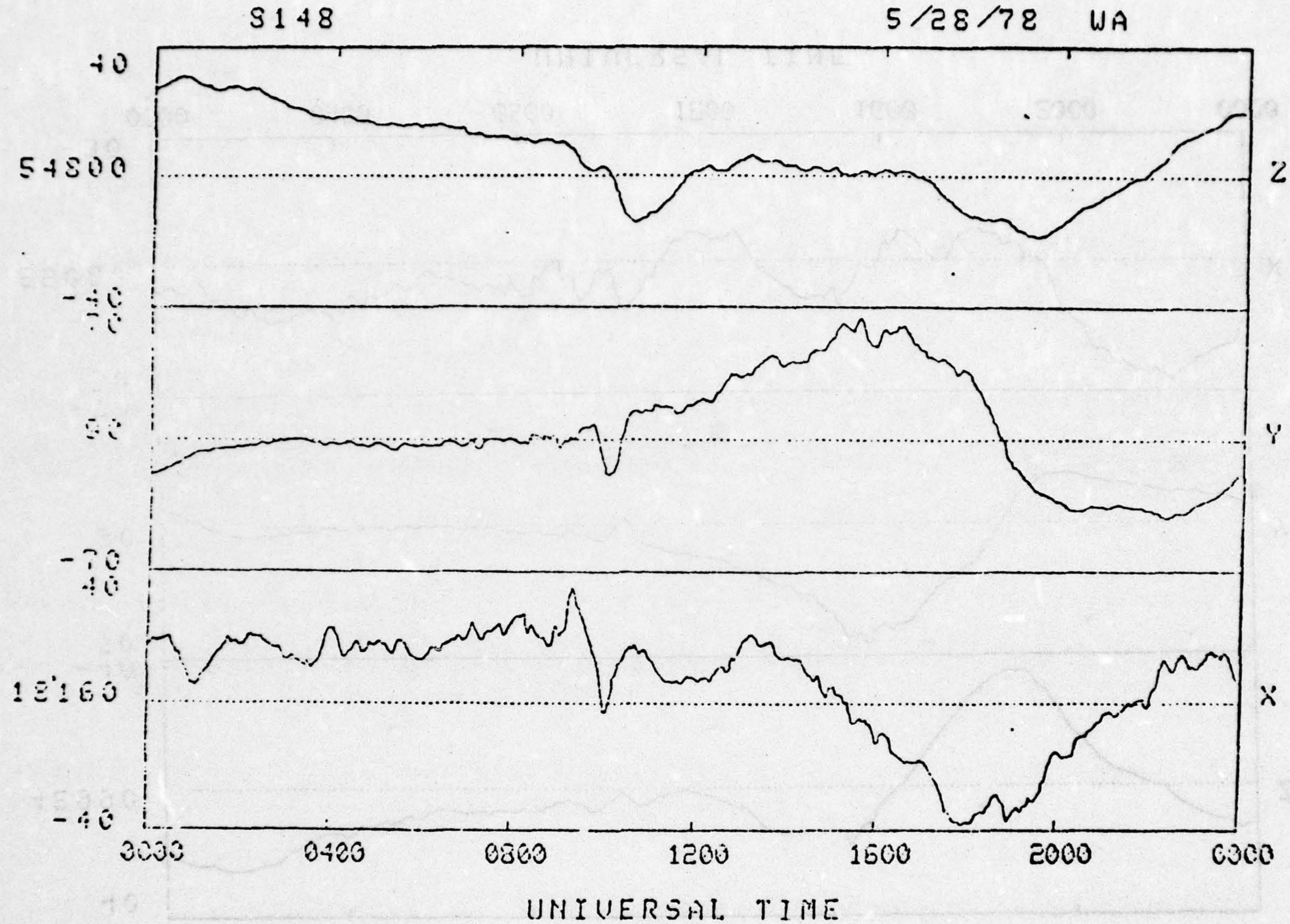
9 April 78
15 April 78
16 April 78
27 May 78
28 May 78

Data have been taken from the Varian System and inputted to the CDC 6600 System at AFGL.

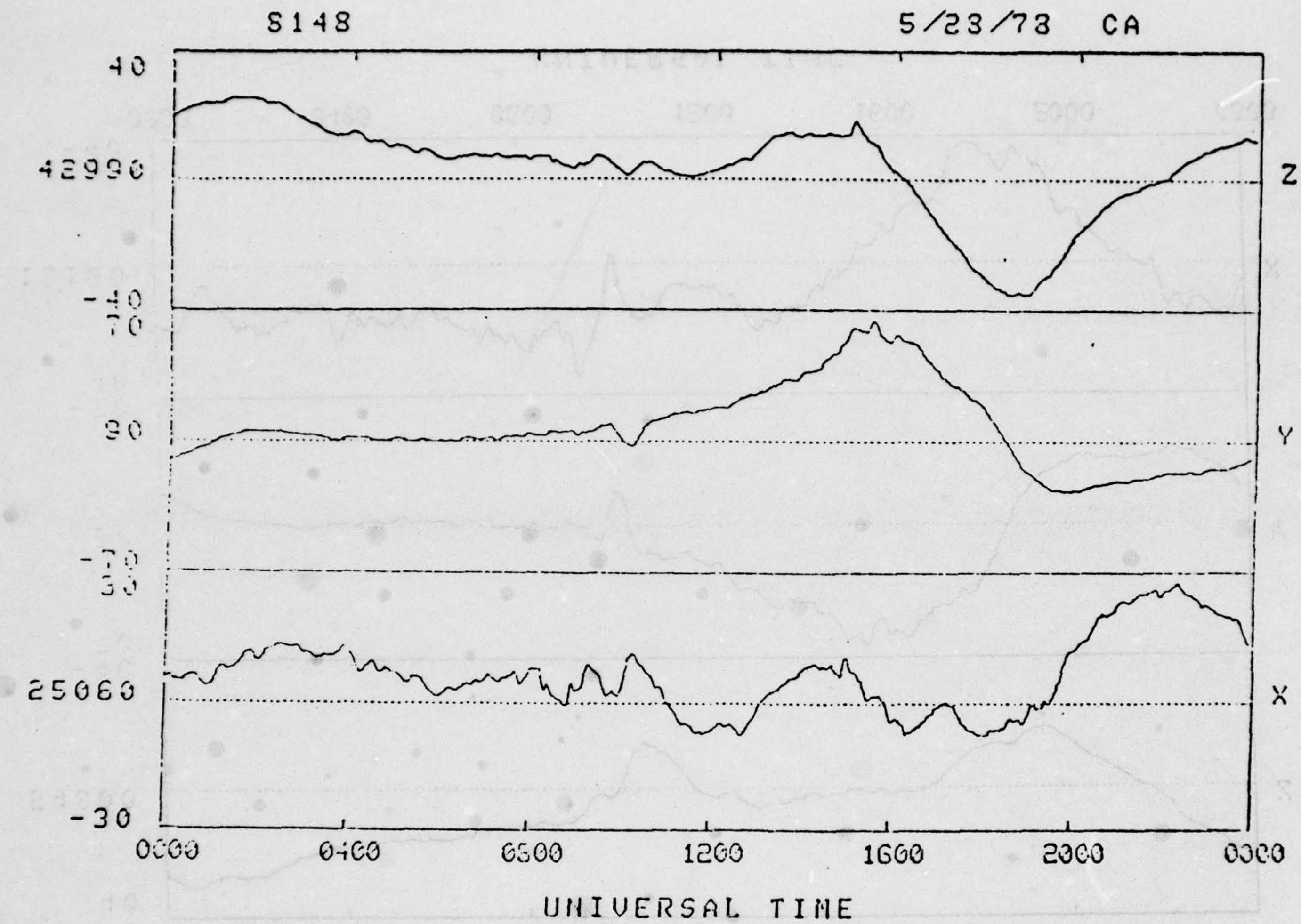
Modifications have been made to the existing program for filtering the data; this filtering is now in progress.

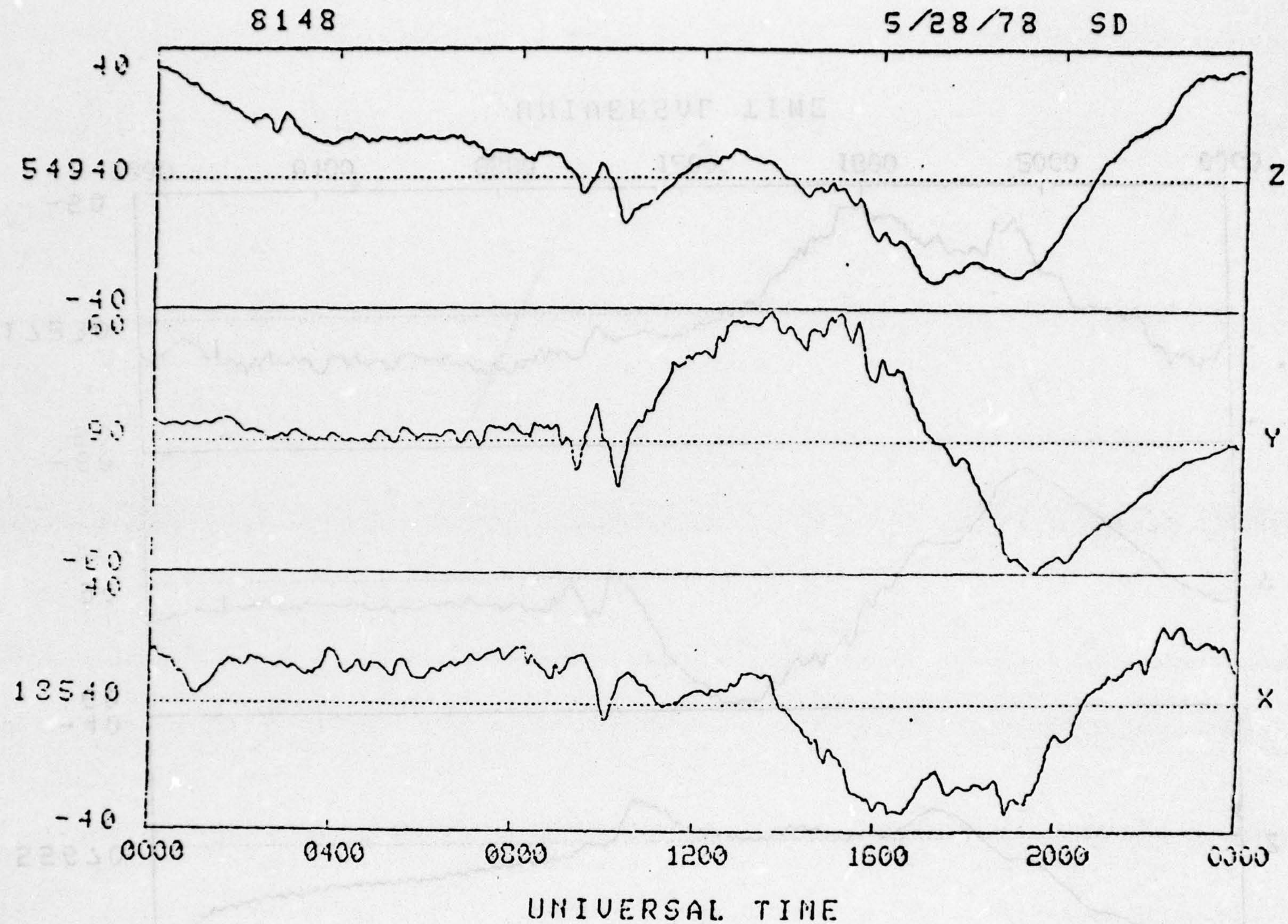
Maximum entropy program will be used to obtain the frequency spectrum of the filtered data.

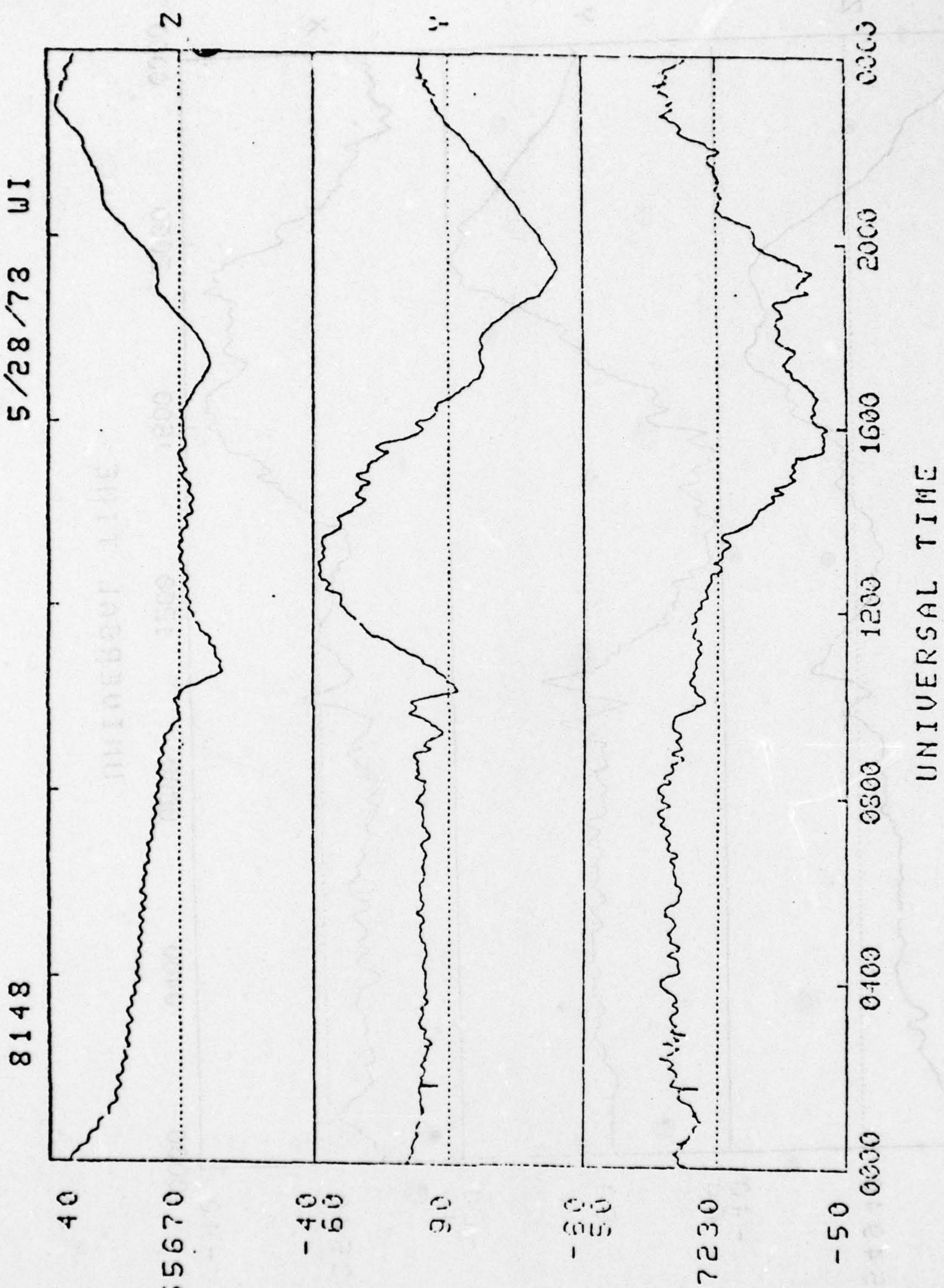
After obtaining the frequency spectrum, results will be presented in the form of a dynamic spectrum.

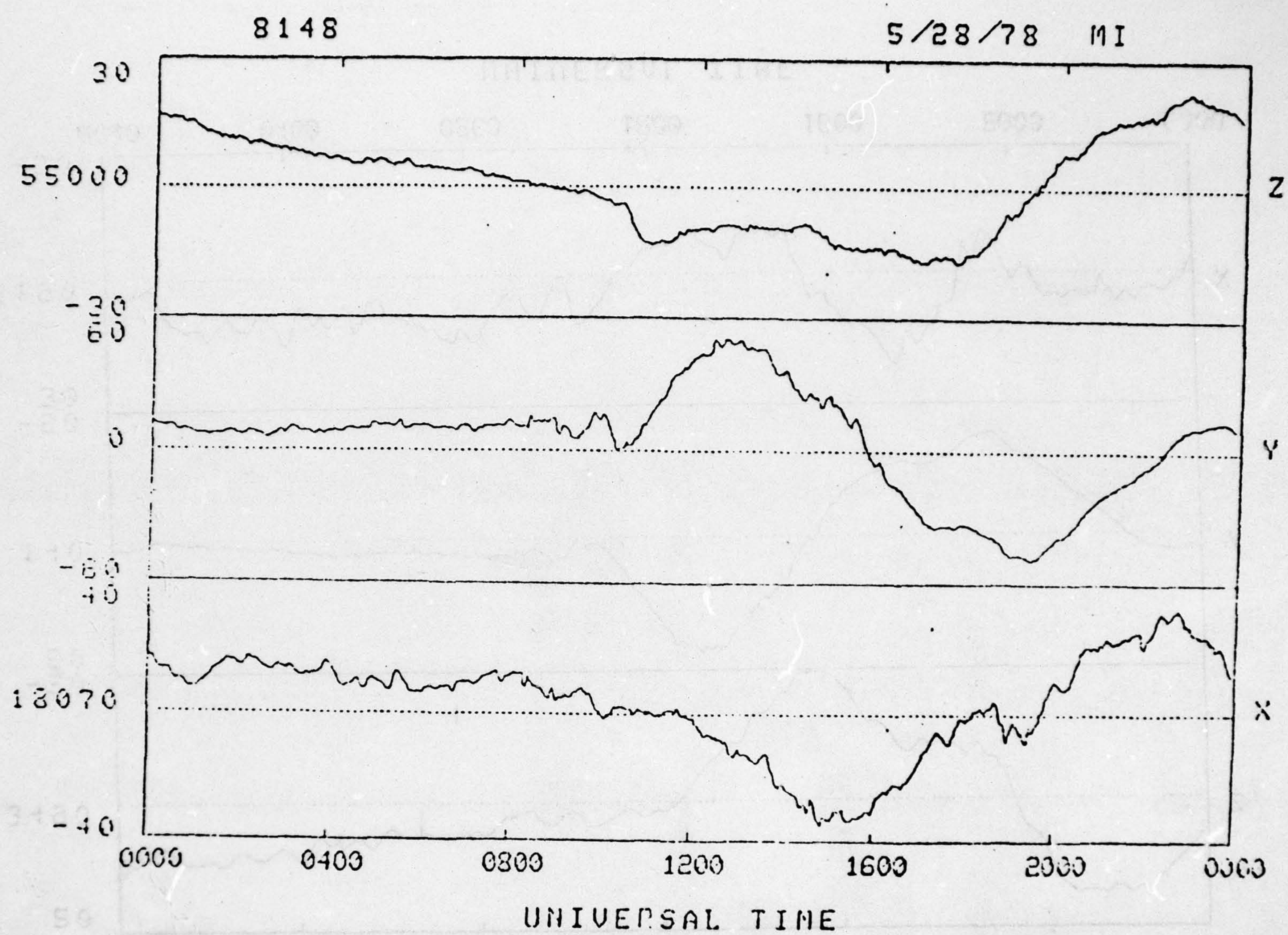


S148
5/28/78 WA
40
54800
-40
70
Z
90
Y
-70
40
18160
-40
X
0000
0400
0800
1200
1600
2000
0000
UNIVERSAL TIME

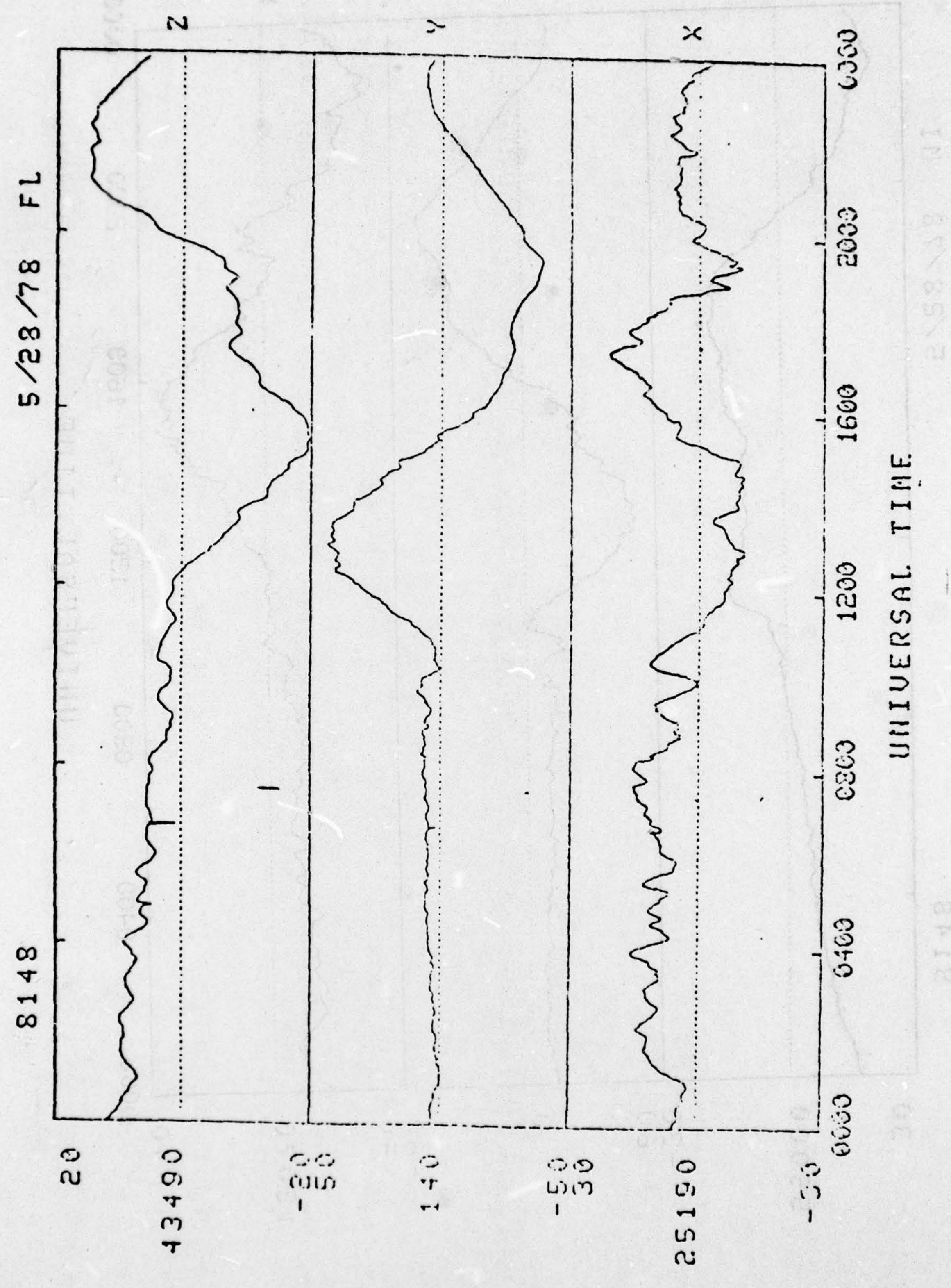

S148
5/23/73 CA
40
42990
-40
70
90
-70
30
25060
-30
Z
Y
X
0000
0400
0800
1200
1600
2000
0000
UNIVERSAL TIME

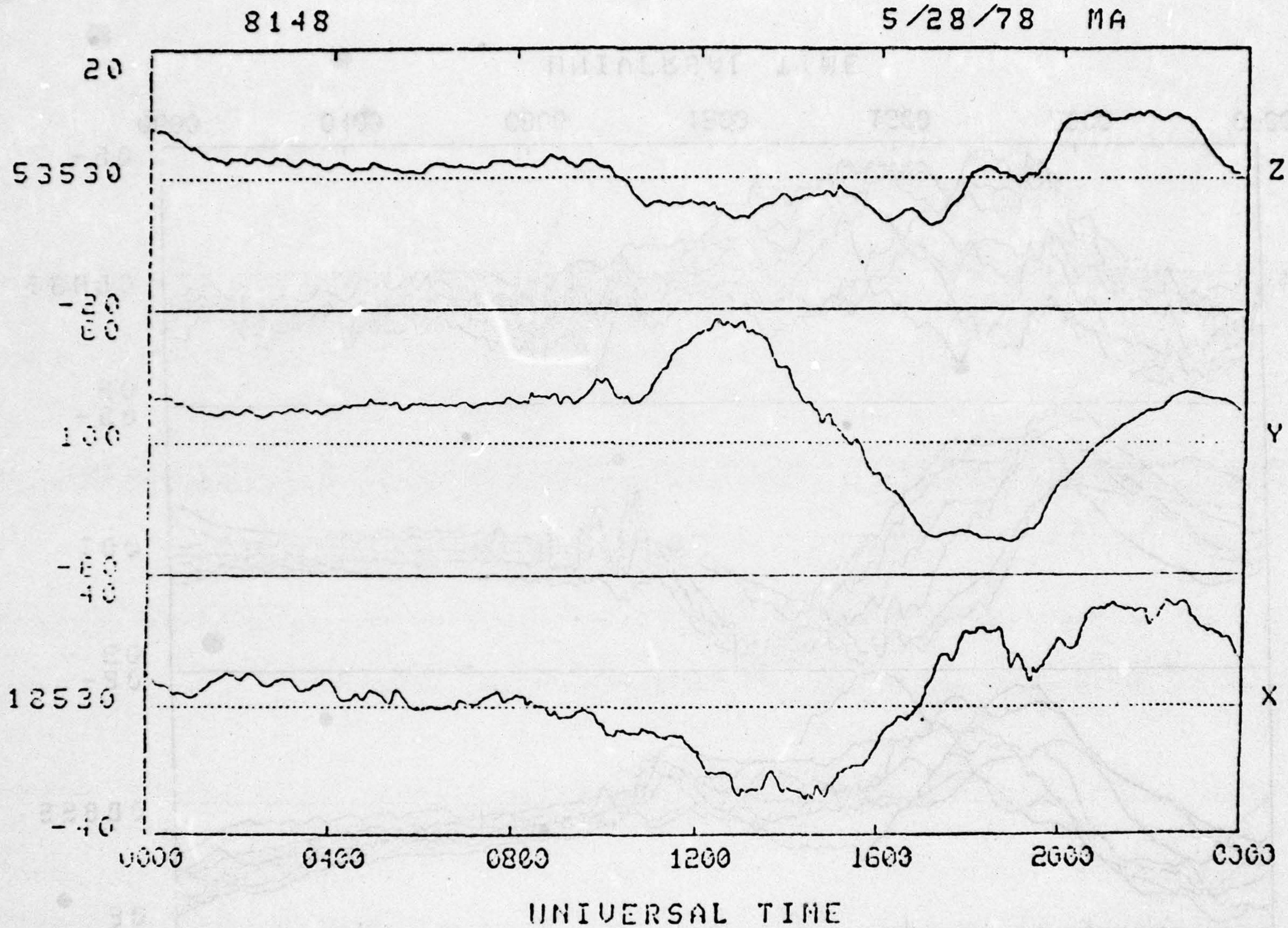

8148
5/28/78 SD
40
54940
-40
60
90
-50
40
13540
-40
Z
Y
X
0000
0400
0800
1200
1600
2000
0000
UNIVERSAL TIME

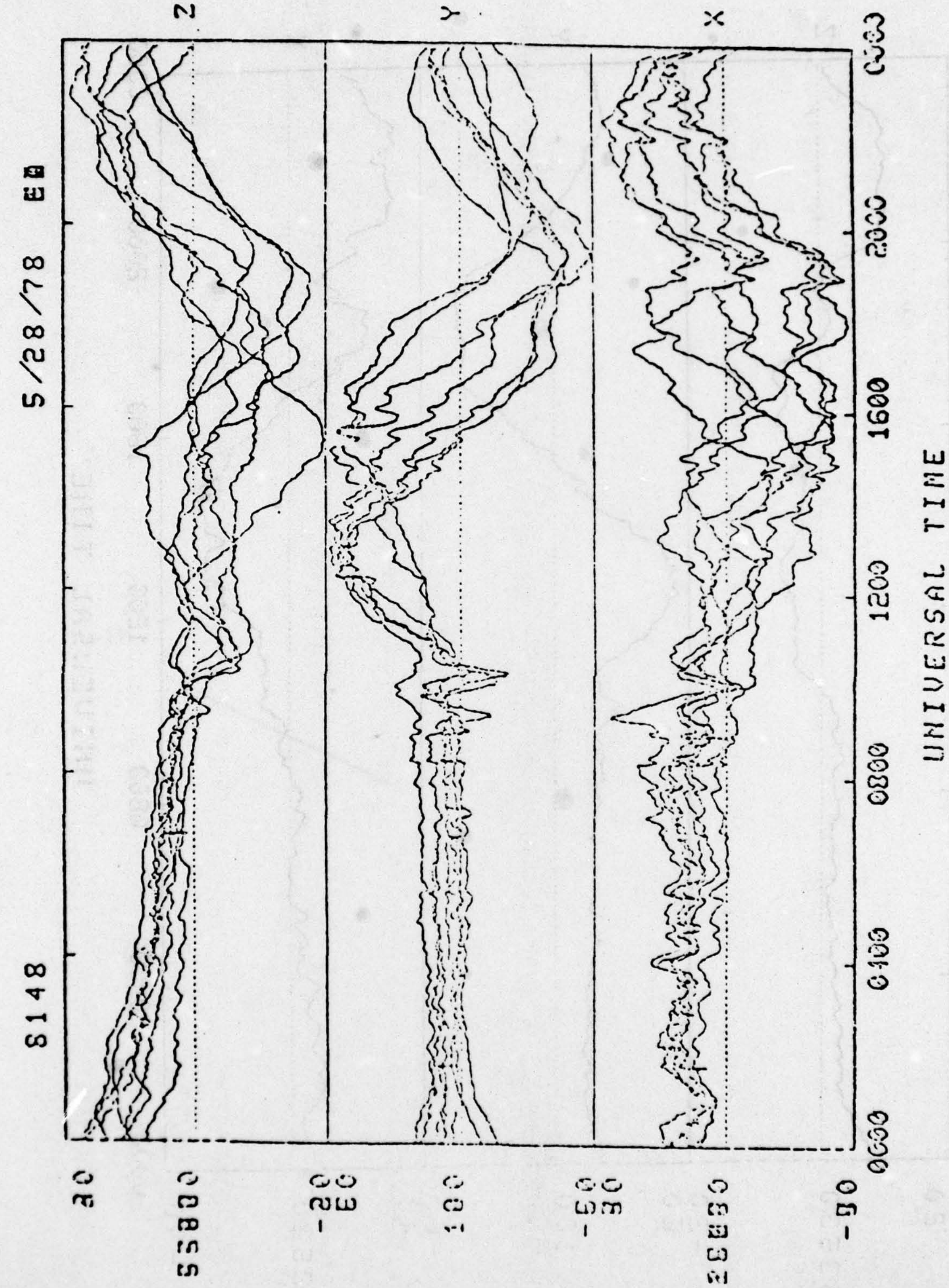

8148
5/28/73 WI
Z
Y
40
55670
-40
60
90
-80
50
17230
-50
0000
0400
0800
1200
1600
2000
UNIVERSAL TIME



8148
5/28/78 MI
30
55000
-30
60
0
-60
40
18070
-40
Z
Y
X
0000
0400
0800
1200
1600
2000
0000
UNIVERSAL TIME


8148
5/28/78 FL
Z
Y
X
20
43490
-20
50
140
-50
30
25190
-30
0000
0400
0800
1200
1600
2000
UNIVERSAL TIME


8148
5/28/78 MA
20
53530
-20
60
100
-60
40
12530
-40
Z
Y
X
0000
0400
0800
1200
1600
2000
0000
UNIVERSAL TIME



S148
5/28/78 EB
UNIVERSAL TIME
0000
0400
0800
1200
1600
2000
Z
Y
X

6. LONGITUDINAL EXTENT - GEOMAGNETIC PULSATIONS

A search was carried out of the network data on the 5 quietest days of the month up to December 1977. From this search 9 days were selected as promising: 4 May, 27 Apr, 26 May, 27 July, 5 Sept, 9 Oct, 23 Nov, 24 Nov, 8 Dec and 18 Dec. However these proved unsatisfactory because of gaps in the data or noisy data. The search will be resumed for the period when all seven stations are operating.

A talk was given at the spring meeting of the AGU. The abstract is attached below.

LONGITUDINAL EXTENT OF OCCURRENCE OF Pc4 AND Pc5 GEOMAGNETIC PULSATIONS

W. Bellew (Physics Research Division, Emmanuel College, Boston MA 02115)
V.L. Patel* (Dept. of Physics, Massachusetts Institute of Technology, Cambridge MA 02139)
P. Fougere (Air Force Geophysics Laboratory, Hanscom AFB MA 01731)

We have analyzed magnetic pulsation events from the AFGL magnetometer network located at $\sim$55°N geomagnetic latitude and covering 3 hours of local time sector. Pulsation events with period greater than 80s have been studied on June 3, 1977. The results indicate that the pulsations of Pc4 with small amplitudes (few gammas) are localized in a longitudinal sector of 1 or 2 hours extent. However, the pulsations with large amplitudes ($\gtrsim 10\gamma$) and longer periods are observed over a longitudinal extent of 3 hours. Detailed results of wave characteristics will be presented and will be compared with theoretical models for the pulsations.

1. 023872 FOUGERE
2. 1978 Spring Meeting
3. Solar-Planetary Relations
4. Micropulsations
5. No
6. No
7. None
8. Bill to:
   Emmanuel College
   Physics Research Div.
   400 The Fenway
   Boston MA 02115
9. 1143